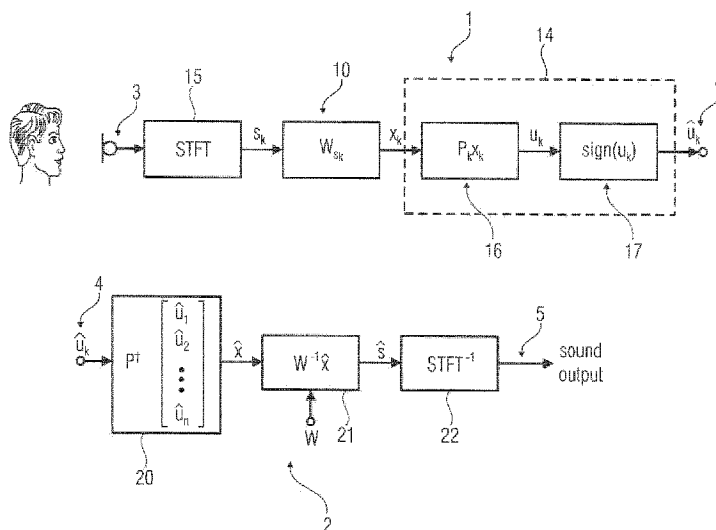




(86) Date de dépôt PCT/PCT Filing Date: 2017/03/10
(87) Date publication PCT/PCT Publication Date: 2017/09/21
(45) Date de délivrance/Issue Date: 2021/09/28
(85) Entrée phase nationale/National Entry: 2018/09/11
(86) N° demande PCT/PCT Application No.: EP 2017/055716
(87) N° publication PCT/PCT Publication No.: 2017/157800
(30) Priorités/Priorities: 2016/03/15 (EP16160369.1);
2016/09/21 (EP16189893.7)

(51) Cl.Int./Int.Cl. *G10L 19/038* (2013.01)
(72) Inventeurs/Inventors:
BAECKSTROEM, TOM, FI;
GHIDO, FLORIN, DE;
FISCHER, JOHANNES, DE
(73) Propriétaire/Owner:
FRAUNHOFER-GESELLSCHAFT ZUR FOERDERUNG
DER ANGEWANDTEN FORSCHUNG E.V., DE
(74) Agent: PERRY + CURRIER

(54) Titre : APPAREIL DE CODAGE PERMETTANT DE TRAITER UN SIGNAL D'ENTREE ET APPAREIL DE
DECODAGE PERMETTANT DE TRAITER UN SIGNAL CODE
(54) Title: ENCODING APPARATUS FOR PROCESSING AN INPUT SIGNAL AND DECODING APPARATUS FOR
PROCESSING AN ENCODED SIGNAL



(57) Abrégé/Abstract:

The invention refers to an apparatus (1) for processing an input signal (3), comprising a perceptual weighter (10) and a quantizer (14). The perceptual weighter (10) comprises a model provider (12) and a model applicator (13). The model provider (12) provides a perceptual weighted model (W) based on the input signal (3). The model applicator (13) provides a perceptually weighted spectrum (x) by applying the perceptual weighted model (W) to a spectrum (s) based on the input signal (3). The quantizer (14) is configured to quantize the perceptually weighted spectrum (x) and for providing a bitstream (fi). The quantizer (14) comprises a random matrix applicator (16) and a sign function calculator (17). The random matrix applicator (16) is configured for applying a random matrix (P) to the perceptually weighted spectrum (x) in order to provide a transformed spectrum (u). The sign function calculator (17) is configured for calculating a sign function of components of the transformed spectrum (u) in order to provide the bitstream (u). The invention further refers to an apparatus (2) for processing an encoded signal (4) and to corresponding methods.

(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property
Organization
International Bureau

(43) International Publication Date
21 September 2017 (21.09.2017)



(10) International Publication Number
WO 2017/157800 A1

(51) International Patent Classification:
G10L 19/038 (2013.01)

(21) International Application Number:
PCT/EP2017/055716

(22) International Filing Date:
10 March 2017 (10.03.2017)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
16160369.1 15 March 2016 (15.03.2016) EP
16189893.7 21 September 2016 (21.09.2016) EP

(71) Applicant: **FRAUNHOFER-GESELLSCHAFT ZUR FÖRDERUNG DER ANGEWANDTEN FORSCHUNG E.V.** [DE/DE]; Hansastraße 27c, 80686 München (DE).

(72) Inventors: **BÄCKSTRÖM, Tom**; Hietaniemenkatu 1 E 13, 00100 Helsinki (FI). **GHIDO, Florin**; Am Stadtpark 21, 90409 Nürnberg (DE). **FISCHER, Johannes**; Juraquelle 18, 91330 Bammsersdorf (Eggolsheim) (DE).

(74) Agents: **ZINKLER, Franz** et al.; Schoppe, Zimmermann, Stöckeler, Zinkler, Schenk & Partner mbB, Radtkoferstr. 2, 81373 München (DE).

(81) Designated States (unless otherwise indicated, for every kind of national protection available): AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BN, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DJ, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IR, IS, JP, KE, KG, KH, KN, KP, KR, KW, KZ, LA, LC, LK, LR, LS, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PA, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SA, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.

(84) Designated States (unless otherwise indicated, for every kind of regional protection available): ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, ST, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, RU, TJ, TM), European (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, KM, ML, MR, NE, SN, TD, TG).

Published:

— with international search report (Art. 21(3))

(54) Title: ENCODING APPARATUS FOR PROCESSING AN INPUT SIGNAL AND DECODING APPARATUS FOR PROCESSING AN ENCODED SIGNAL

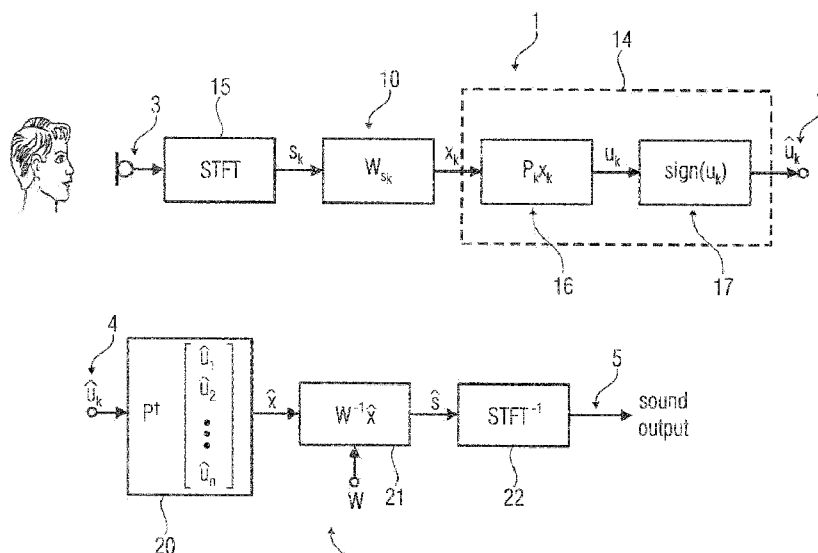


Fig. 6

(57) Abstract: The invention refers to an apparatus (1) for processing an input signal (3), comprising a perceptual weighter (10) and a quantizer (14). The perceptual weighter (10) comprises a model provider (12) and a model applicator (13). The model provider (12) provides a perceptually weighted model (W) based on the input signal (3). The model applicator (13) provides a perceptually weighted spectrum (x) by applying the perceptually weighted model (W) to a spectrum (s) based on the input signal (3). The quantizer (14) is configured to quantize the perceptually weighted spectrum (x) and for providing a bitstream (u). The quantizer (14) comprises a random matrix applicator (16) and a sign function calculator (17). The random matrix applicator (16) is configured for applying a random matrix (P) to the perceptually weighted spectrum (x) in order to provide a transformed spectrum (u). The sign function calculator (17) is configured for calculating a sign function of components of the transformed spectrum (u) in order to provide the bitstream (u). The invention further refers to an apparatus (2) for processing

an encoded signal (4) and to corresponding methods.

Encoding apparatus for processing an input signal and decoding apparatus for processing an encoded signal

Specification

5

The invention refers to an encoding apparatus for processing an input signal and to a decoding apparatus for processing an encoded signal. The invention also refers to corresponding methods and to a computer program.

- 10 A central part of speech and audio codecs are their perceptual models, which describe the relative perceptual importance of errors in different elements of the signal representation. In practice, the perceptual models consist of signal-dependent weighting factors which are used in quantization of each element. For optimal performance, it would be desirable to use the same perceptual model at the decoder. While the perceptual model is signal-
- 15 dependent, however, it is not known in advance at the decoder, whereby audio codecs generally transmit this model explicitly, at the cost of increased bit-consumption.

The era of Internet of Things (IoT) is approaching, whereby the next generation of speech and audio coders should embrace it. The design goals of IoT-systems however fit poorly

20 with the classic design of speech and audio coders, whereby a larger redesign of the coders is required.

Primarily, whereas state-of-the-art speech and audio coder such as AMR-WB, EVS, USAC and AAC consist of intelligent and complex encoders and relatively simple

25 decoders [1 – 4], since IoT should support distributed low-complexity sensor-nodes, encoders are preferred to be simple.

Secondly, since sensor-nodes are encoding the same source signal, application of the same quantization at each sensor-node would represent over-coding and potentially a

30 serious loss in efficiency. Especially, since the perceptual model should be more or less the same at every node, transmitting it from every node is almost pure over-coding.

Conventional speech and audio coding methods consist of three parts:

- 35 1. a perceptual model which specifies the relative impact of errors in different parameters of the codec,

2. a source model which describes the range and likelihood of different inputs and
3. an entropy coder which utilizes the source model to minimize perceptual distortion [5].

5 Further, the perceptual model can be applied in either of two ways:

1. Signal parameters can be weighted according to the perceptual model, such that all parameters can then be quantized with the same accuracy. The perceptual model must then be transmitted to the decoder such that the weighting can be
10 undone.
2. The perceptual model can alternatively be applied as an evaluation model, such that the synthesis output of different quantizations are compared, weighted by the perceptual model, in an analysis-by-synthesis iteration. Though here the perceptual model has not to be transmitted, this approach has the disadvantage
15 that quantization cells shapes are not regularly shaped which reduces coding efficiency. More importantly, however, to find the optimal quantization, a computationally complex brute-force search of different quantizations has to be used.

20 Since the analysis-by-synthesis approach thus leads to a computationally complex encoder, it is not a viable alternative for IoT. Therefore, the decoder must have access to the perceptual model. However, as noted above, explicit transmission of the perceptual model (or equivalently, an envelope model of the signal spectrum), is not desirable because it lowers coding efficiency.

25

The object of the invention is to present a way to recover the perceptual model at the decoder from the transmitted signal without side-information concerning the perceptual model.

30 The object is achieved by an encoding apparatus for processing an input signal, as well as by a decoding apparatus for processing an encoded signal. The object is also achieved by corresponding methods and a system comprising the encoding apparatus and the decoding apparatus.

The invention is especially useful in distributed sensor-networks and the Internet of things, where the added cost on bit-consumption from transmitting a perceptual model increases with the number of sensors.

- 5 The invention according to one embodiment might also be labelled as a blind recovery of perceptual models in distributed speech and audio coding.

The input signal here is a speech and/or an audio signal to be encoded by the encoding apparatus.

10

The object is achieved by an encoding apparatus for processing an input signal. The encoding apparatus preferably provides an encoded signal.

The encoding apparatus comprises a perceptual weighter and a quantizer.

15

The perceptual weighter comprises a model provider and a model applicator. The model provider is configured for providing a perceptual weighted model based on the input signal. The model applicator is configured for providing a perceptually weighted spectrum by applying the perceptual weighted model to a spectrum based on the input signal.

20

The quantizer is configured to quantize the perceptually weighted spectrum and for providing a bitstream. The quantizer comprises a random matrix applicator and a sign function calculator. The random matrix applicator is configured for applying a random matrix to the perceptually weighted spectrum in order to provide a transformed spectrum.

- 25 The sign function calculator is configured for calculating a sign (or: signum) function of components of the transformed vector in order to provide the bitstream.

The quantization comprises at least two steps: In a first step, the perceptually weighted spectrum is combined with a random matrix. Such a random matrix has the advantage that for each input signal a different matrix is used. This becomes useful when a plurality of sensors covers the same audio source and over-coding has to be avoided. The second step includes that the sign function is calculated of the components of the transformed vector.

30

In one embodiment, the model provider is configured for providing the perceptual weighted model based on a compression of the spectrum based on the input signal.

35

In the inventive encoding apparatus the perceptual weighting of the input signal is based in one embodiment on a compression of the input signal. In an embodiment, the perceptual weighting is based on the compression of the envelope of the magnitude spectrum of the input signal (an envelope is a continuous, usually smooth shape describing a characteristic of the signal, her of the magnitude spectrum). Based on the compression a perceptual weighted model is obtained which is finally used to perceptually weight the spectrum of the input signal.

10 The encoding apparatus provides in one embodiment a bitstream with side-information covering some aspects about the encoding process.

In an embodiment, information about the quantization is provided as side-information by the encoded signal output by the encoding apparatus as result of the encoding process.

15

In a further embodiment, the perceptual weighter comprises an envelope calculator. The envelope calculator is configured for providing an envelope of a magnitude spectrum based on the input signal.

20 In one embodiment, the model provider is configured for calculating a compression function describing the compression of the envelope. Further, the model provider is configured for calculating the perceptual weighted model based on the compression function. In this embodiment, the model provider compresses the envelope and calculates a function describing this compression. Based on this function, the perceptual weighted
25 model is obtained. In this embodiment, a compression of an envelope of a spectrum based on the input signal is performed, i.e. the magnitude-range is reduced and, thus, smaller than before the compression. By compressing the envelope, the spectrum as such is also compressed, i.e. the magnitude-range of the spectrum is reduced.

30 In a different embodiment, the perceptual weighted model or the compression function is calculated directly from the input signal or the magnitude/power spectrum based on the input signal.

35 According to an embodiment, the model provider is configured for calculating a compression function describing a compression of the spectrum based on the input signal or describing a compression of an envelope of a magnitude spectrum based on the input

signal. The compression reduces a magnitude-range of the spectrum based on the input signal or reduced the magnitude-range of the envelope. Further, the model provider is configured for calculating the perceptual weighted model based on the compression function.

5

In an embodiment, the compression function – for compressing e.g. the spectrum or the envelope – fulfils two criteria:

At first, the compression function is strictly increasing. This implies that for any positive scalar value and an arbitrarily small value the function value for the positive scalar value is smaller than the function value for the sum of the positive scalar value and of the

10

arbitrarily small value.

At second, for a first positive scalar value and a second positive scalar value greater than the first positive scalar value, a difference between the function value for the second positive scalar value and the function value for the first positive scalar value is smaller than a difference between the second positive scalar value and the first positive scalar value.

15

The object is also achieved by a method for processing an input signal. This input signal is preferably an audio signal and/or a speech signal.

20

The method for processing an input signal – being an audio and/or speech signal – comprises at least the following steps:

calculating a perceptual weighted model based on the input signal,

25

providing a perceptually weighted spectrum by applying the perceptual weighted model to a spectrum based on the input signal, and

quantizing the perceptually weighted spectrum in order to provide a bitstream,

wherein quantizing the perceptually weighted spectrum comprises:

applying a random matrix to the perceptually weighted spectrum in order to provide a transformed spectrum, and

30

calculating a sign function of components of the transformed spectrum in order to provide the bitstream.

In an embodiment, the following steps are performed:

35

calculating an envelope of a magnitude spectrum based on the input signal,

calculating the perceptual weighted model based on a compression of the envelope.

The embodiments of the encoding apparatus can also be performed by steps of the method and corresponding embodiments of the method. Therefore, the explanations given for the embodiments of the apparatus also hold for the method.

5

The object is further achieved by a method for processing an input signal, comprising:

providing a perceptual weighting model based on the input signal,

weighting a spectrum of the input signal by applying the perceptual weighting model to the spectrum of the input signal, and

10

quantizing the weighted spectrum by calculating a sign function of random projections of the weighted spectrum.

The method further comprises in an embodiment:

15

obtaining the random projections of the weighted spectrum by applying a random matrix to the weighted spectrum.

In an embodiment, providing the perceptual weighting model comprises compressing an envelope of a magnitude spectrum of the input signal.

20

According to an embodiment, the method further comprises:

obtaining the envelope by using a filterbank and a diagonal matrix containing normalization factors for each band.

25

The object is also achieved by a decoding apparatus for processing an encoded signal.

The decoding apparatus comprises at least a de-quantizer and a perceptual de-weighter.

30

The de-quantizer is configured to de-quantize a bitstream comprised by the encoded signal and for providing a calculated perceptually weighted spectrum. Further, the de-quantizer is configured to de-quantize the bitstream by applying a pseudo-inverse of a random matrix to the bitstream. The de-quantizer reverses effects of a quantization happened during the encoding process. The de-quantizer is followed by the perceptual de-weighter such that the spectrum resulting from the de-quantization is perceptually de-weighted. The calculated perceptually weighted spectrum is received by the perceptual

35

de-weighter in order to be perceptually de-weighted. The finally obtained spectrum is, thus, a perceptually de-weighted de-quantized bitstream comprised by the input signal.

5 The perceptual de-weighter reverses the effects of a perceptual weighting that happened during the encoding process leading to the encoded signal. This is done in one embodiment without side-information of the encoded signal comprising the perceptual weighted model. The model is reconstructed from the encoded audio signal as such.

10 The perceptual de-weighter comprises a spectrum approximator and a model approximator.

15 The reconstruction of the model is done in one embodiment in an iterative way for which starting or initial values are required. Hence, an initial guess provider comprised by the perceptual de-weighter is configured for providing data for an initial guess of a perceptual weighted model with which the encoded signal is associated. The data for the initial guess comprises in one embodiment a vector with the diagonal elements of a matrix describing the perceptual weighted model.

20 The spectrum approximator is configured for calculating an approximation of a spectrum based on the calculated perceptually weighted spectrum. Further, the model approximator is configured for calculating an approximation of a perceptual weighted model with which the encoded signal is associated (i.e. which was used for encoding of the input signal and, thus, generating the encoded signal) based on the approximation of a spectrum.

25 In an embodiment, the perceptual de-weighter comprises an initial guess provider. The initial guess provider is configured for providing data for an initial guess of the perceptual weighted mode. The spectrum approximator is configured for calculating the approximation of the spectrum based on the encoded signal and the initial guess or the approximation of the perceptual weighted model. Further, the approximation of a spectrum
30 is based in an embodiment either on the initial guess or on an – especially calculated – approximation of the perceptual weighted model. This choice depends on the fact whether the iteration just has started – with the initial guess – or at least one iteration with at least one improvement of the approximation – leading to the approximation of the perceptual weighted model – has happened already. The iteration is performed in one embodiment
35 until a convergence criterion has been fulfilled.

The de-quantizer is configured in one embodiment to de-quantize the bitstream comprised by the encoded signal based on side-information about the random matrix comprised by the encoded signal. This embodiment refers to the encoding process in which a quantization is performed using a random matrix. The information about the used random matrix is comprised by the side-information of the encoded signal. The side-information comprises in one embodiment just the seed of the columns of the random matrix.

In one embodiment, a plurality of encoded signals is processed jointly. Each encoded signal comprises a bitstream with side-information concerning at least information about the quantization performed while providing the respective encoded signal. For this purpose, the de-quantizer is configured to receive a plurality of input signals and to provide a and in one embodiment just one calculated perceptually weighted spectrum based on the plurality of input signals. The input signals preferably refer to the audio/speech signal stemming from the same signal source.

15

In another embodiment, the decoding apparatus is configured to apply source modelling. A source model describes the range and likelihood of different inputs.

The object is also achieved by a method for processing an encoded signal. This encoded signal is preferably an encoded audio signal and/or an encoded speech signal.

20

The method for processing (or decoding) an encoded signal comprises at least the following steps:

de-quantizing a bitstream comprised by the encoded signal and providing a calculated perceptually weighted spectrum,
wherein de-quantizing the bitstream comprises applying a pseudo-inverse of a random matrix to the bitstream,
calculating an approximation of a spectrum based on the calculated perceptually weighted spectrum, and
calculating an approximation of a perceptual weighted model with which the encoded signal is associated (i.e. which was used for generating the encoded signal) based on the approximation of a the spectrum.

30

In one embodiment, the method comprises the steps:

35

calculating an approximation of the spectrum either based on the calculated perceptually weighted spectrum and an initial guess or based on the calculated perceptually weighted spectrum and an approximation of the perceptual weighted model with which the encoded signal is associated.

5

The calculated approximations of the perceptual weighted model are preferably used for the next calculation of the approximation of a spectrum.

The object is also achieved by a method for processing an encoded signal,

10

comprising:

providing a quantized perceptual signal by applying a pseudo-inverse of a random matrix to the encoded signal,

calculating an estimation of a spectrum based on the quantized perceptual signal, and

calculating an approximation of a perceptual weighting model used for providing the

15

encoded signal based on the approximation of the spectrum.

According to an embodiment, the method further comprises:

providing a zero-th approximation of the perceptual weighting model using an initial guess, and

20

calculating a zero-th estimation of the spectrum based on the zero-th approximation of the perceptual weighting model.

In a further embodiment, the method further comprises:

obtaining the initial guess by using a filterbank and a diagonal matrix containing normalization factors for each band.

25

The embodiments of the apparatus can also be performed by steps of the method and corresponding embodiments of the method. Therefore, the explanations given for the embodiments of the apparatus also hold for the method.

30

The object is also achieved by a system comprising at least one encoding apparatus and a decoding apparatus. In one embodiment, a plurality of encoding apparatuses are used which in one embodiment are associated with sensor-nodes, e.g. microphones.

35

The object is also achieved by a computer program for performing, when running on a computer or a processor, the method of any of the preceding embodiments.

The invention will be explained in the following with regard to the accompanying drawings and the embodiments depicted in the accompanying drawings, in which:

5 Fig. 1 shows a block diagram of a first embodiment system comprising an encoding apparatus and a decoding apparatus,

Fig. 2 shows a block diagram of an embodiment of an encoding apparatus,

10 Fig. 3 shows a block diagram of a perceptual weighter as a part of an encoding apparatus,

Fig. 4 shows signals belonging to an encoding process,

15 Fig. 5 shows a block diagram of a second embodiment of a system comprising a multitude of encoding apparatuses,

Fig. 6 shows a more detailed block diagram of a third embodiment of a system,

20 Fig. 7 shows a block diagram of an embodiment of a decoding apparatus,

Fig. 8 shows a block diagram of a different embodiment of a decoding apparatus,

25 Fig. 9 shows an embodiment of a perceptual de-weighter as part of a decoding apparatus,

Fig. 10 shows signals belonging to a decoding process,

30 Fig. 11 shows mean SNR values with different bit-rates for experimental data and

Fig. 12 shows differential MUSHRA scores for experimental data.

Fig. 1 shows a system for handling an input signal 3 which is a speech and/or audio signal. This input signal 3 is encoded by an encoding apparatus 1. The encoded signal 4
35 being preferably a bit stream is, for example, transmitted via the Internet to a decoding

apparatus 2 which decodes the encoded signal 4 and provides an extracted audio signal 5 to a – not shown – listener.

5 The encoding apparatus 1 uses a perceptual weighted model for processing the input signal 3 but this model is not transmitted by the encoded signal 4. The decoding apparatus 2 extracts the model from the encoded signal 4 in order to undo the effects of the model.

10 The encoding apparatus 1 shown in Fig. 2 comprises a transformer 15, a perceptual weighter 10, and a quantizer 14.

The transformer 15 provides a spectrum s based on the input signal 3 being a time signal. This is done, for example, by short-time Fourier transform (STFT).

15 The spectrum s undergoes a perceptual weighting by the perceptual weighter 10 becoming a perceptually weighted spectrum x . This spectrum x is submitted to the quantizer 14 which quantizes it and provides a quantized signal $\hat{u}$ being a bitstream. That quantized signal $\hat{u}$ is in this embodiment combined with corresponding side-information covering information about the quantization – but in this embodiment not about the
20 perceptual weighted model – to provide the encoded signal 4.

Fig. 3 shows an embodiment of the perceptual weighter 10.

25 The perceptual weighter 10 receives the input signal 3 – or a corresponding spectrum after a transformation of the input signal 3 to the frequency domain – and provides a perceptually weighted spectrum x . For this purpose, the perceptual weighter 10 comprises an envelope calculator 11, a model provider 12, and a model applicator 13.

30 The envelope calculator 11 receives the input signal 3 or a corresponding spectrum and provides an envelope y of a magnitude spectrum $|x|$ based on the input signal 3. In the shown embodiment, the envelope calculator 11 provides the envelope y by a matrix A and a diagonal matrix Λ . The matrix A is a filterbank and the diagonal matrix Λ contains normalization factors for each band of the used filterband. The envelope y is then obtained based on the magnitude spectrum $|x|$ of the input signal 3 by the equation
35 $y = A\Lambda A^T|x|$, whereas A^T is the transposed of the matrix A .

Based on this envelope y , the model provider 12 compresses the envelope y . The purpose of the compression is to obtain a function which approximates perceptual performance of the ear. In the following, an embodiment is discussed in which the envelope is compressed by raising y to the power of p . When, for example, p equals 0.3, then the range of y^p will be smaller than the range of the original y . The amount of compression, in this example, thus depends on the value p . For example, the envelope is reduced to a desired range. In one embodiment, a range-reduction or a compression function for compressing the envelope y is given by y^p with $0 < p < 1$. This means that the compression is performed by an exponential function of the envelope with an exponent greater than zero and smaller than one. This compression is performed in one embodiment sample-wise.

The compression function $f(y)$ describes a vector w , i.e. $w = f(y)$ that gives diagonal elements of a perceptual weighted model W . Here, the input y is given as a vector and function f is applied on all samples of the vector y , resulting in a vector w . So if the k 'th sample of y is y_k , then the k 'th sample of $w = f(y)$ is $w_k = f(y_k)$.

Hence, based on the compression function, the perceptual weighted model W can be obtained, here in the form of a matrix.

With other words: The envelope of the magnitude spectrum is compressed and from the function describing the compressed envelope, the perceptual weighted model is calculated which is used to perceptually weight the spectrum.

The model applicator 13 applies the perceptual weighted model W to the spectrum s based on the input signal 3. In the shown embodiment, the model applicator 13 applies the matrix of the perceptual weighted model W to a vector based on the spectrum.

The perceptual modelling will now be explained once more:

Speech and audio codecs are based on efficient modelling of human auditory perception. The objective is to obtain such a weighting of quantization errors that optimization of the signal-to-noise ratio in the weighted domain gives the perceptually best possible quality.

Audio codecs operate generally in the spectral domain, where the spectrum of an input frame s can be perceptually weighted with a diagonal matrix W such that the weighted

spectrum $x = Ws$ can be quantized $\hat{x} = [Ws]$, where the brackets $[\quad]$ denotes quantization.

At the decoder, the inverse operation $\hat{s} = W^{-1}\hat{x}$ can be reconstructed.

5

Specifically, the perceptual weighting model consists of two parts:

i) A fixed part corresponding to the limits of perception at different frequency bands. Perceptual models such as Bark- and ERB-scales model the density of frequencies such that the warped axis has uniform perceptual accuracy [17]. However, since an objective is to measure error energy on the warped scale, the magnitude of spectral components can equivalently be scaled such that the computationally complex warping operation can be avoided [18]. This operation is also similar to the pre-emphasis operation applied in speech codecs [1 – 3]. Since this part of the weighting is fixed, it does not need to be explicitly transmitted. It can be applied at the encoder and directly reversed at the decoder.

ii) The signal-adaptive part of the perceptual model corresponds to the frequency-masking properties of perception. Namely, high-energy components of the signal will mask lower energy components and thus render them inaudible, if the two are sufficiently close in frequency [5]. The shape of the frequency-masking curve is thus equal to the shape of the signal envelope, but with a smaller magnitude.

If $|x|$ is the magnitude spectrum of the input signal, its spectral envelope y can be obtained in an embodiment by $y = A\Lambda A^T|x|$, where matrix A is a filterbank such as in Fig. 4 (a).

In difference to the common MFCC-type filterbanks [19], an asymmetric Hann-window type windows is used in one embodiment with an overlap extending from the k th filter to the $(k - 2)$ and $(k + 2)$ filters (see Fig. 4 (a)).

30

The diagonal matrix Λ contains normalization factors for each band such that unit-gain is obtained.

Depending on the embodiment, a Mel-, Bark- or ERB-scale is used with a suitable number of bands.

35

At a sampling rate of 12.8 kHz, here a Mel-filterbank with 20 bands was used.

An alternative to the MFCC-type filterbank matrix is to use spreading by filtering, whereby A becomes a convolution matrix. Since filtering operations are well-understood digital
5 signal processing methods, their inverses are readily found as well.

The perceptual weighting factors model the frequency masking effect, which in turn corresponds to spreading and scaling of energy over frequencies [20, 5]. The envelope model matrix A already achieves the effect of spreading, whereby still the scaling of
10 energy has to be modelled.

The energy scaling corresponds to compression of the signal, which reduces the magnitude-range of the envelope (see Fig. 4 (b)). Hence, if the spectrum s is multiplied with the perceptual weighting matrix W , a spectrum $x = Ws$ is obtained which has a
15 reduced range (see Fig. 4 (c)).

Perceptual weighting, thus, reduces the range or flattens the spectrum, but it does not produce a spectrum with an entirely flat envelope. The range of the envelope is reduced, whereby a part of its range is retained, and that remaining range can be used to recover
20 the original signal following the expanded envelope.

The range-reduction or compression function $w = f(y)$ for the envelope y (where vector w gives the diagonal elements of W), can be applied for example as a sample-wise exponent $f(y) = y^p$ with $0 < p < 1$.
25

Given a spectrum s and its k 'th sample s_k , weighting is applied by multiplication such that $x_k = w_k * s_k$, where x_k is the k 'th sample of the weighted spectrum x and w_k is the k 'th sample of the weighting vector w . The same operation can be expressed as a matrix operation by generating a matrix W , which has the weighting values on the diagonal $W_{kk} =$
30 w_k , and the matrix is zero at all other positions. It follows that $x = W*s$.

While it is possible to use any function which compresses the range of y , exponentiation has the benefit that it leads to simple analytic expression in envelope reconstruction at the decoder.
35

Suitable compression functions $f()$ fulfil the following requirements:

1. the compression functions are strictly increasing, i.e. $f(t) < f(t+\text{eps})$, where t is any positive scalar value and eps is an arbitrarily small value,
2. for any positive scalar values (first and second scalar value, t_1, t_2) $t_1 < t_2$, it holds $f(t_2) - f(t_1) < t_2 - t_1$. In other words, any function which reduces the distance between such two positive scalar values $t_2 - t_1$ is a suitable function.

In addition to exponentiation $f(y) = y^p$, with a small exponent p , in a different embodiment, the compression function is the logarithm, that is, $f(y) = \log(y)$.

10

The encoder algorithm as an encoding method or realized by the encoding apparatus is then in one embodiment:

1. Calculate the envelope of the magnitude spectrum.
2. Compress the envelope to obtain the perceptual weighting model.
3. Apply weighting on spectrum $x = Ws$.
4. Quantize and transmit weighted spectrum $\text{sign}(Px)$.

15

This algorithm is applied independently at every sensor-node.

20

The perceptual weighting is followed by a quantization.

The invention, thus, consists of two parts:

1. distributed quantization of the input signal using random projections and 1 bit quantization, and
2. implicit transmission of the perceptual model.

25

By quantization of random projections, each transmitted bit encodes a unique piece of information and over-coding is avoided.

30

The perceptual model is generated independently at each sensor-node (e.g. comprising a microphone) and the quantized perceptually weighted signal is transmitted. Perceptual weighting makes the signal more flat, but the basic shape is retained. Therefore, it can be inversely deduced at the decoder side what the original envelope must have been, even from the perceptually weighted signal.

35

It follows a discussion of the distributed quantization:

5 Though distributed source coding is a well-studied subject (e.g. [7, 8]) and it has been applied in other applications such as video [9], only a few have worked on distributed audio coding (e.g. [10 – 13]), and none of them however addresses the over-coding problem with respect to perceptual and envelope models. Even the scalable coding approach in [14] includes envelope coding with scale factors. Also the multiple description coding approach has been applied only to packet loss concealment [15, 16].

10

It follows an easily implementable quantization scheme. Compare the 1 bit quantization method which has been used in compressive sensing systems [6].

15 The objective of the quantizer and the quantization process is to allow quantization at independent sensors, such that it is ensured that each transmitted bit improves quality, without communication between sensor-nodes. In the extreme, a sensor could send only one bit and that single bit could be used to improve quality.

20 The proposed quantization scheme of one embodiment is based on random projections of a real-valued representation of the signal spectrum and transmitting the sign of each dimension.

25 Let x be the real-valued N times 1 vector containing the spectrum of the input signal, and P a K times N random matrix whose columns are normalized to unit length. Then, x will be transformed by $u = Px$. This is followed by a quantization of the sign of each component of u , that is, the quantization is $\hat{u} = \text{sign}(u)$, which can be transmitted losslessly with K bits.

30 The number of bits of the bitstream, thus, defines one dimension of the random matrix.

35 The samples of P are preferably pseudo-random values, which means that they look like random values, but are actually generated by some complex mathematical formula or algorithm. Pseudo-random generators are standard mathematical tools which every computer and every mathematical software-library has. Relevant are: that the matrix P has to be known at both the encoder and receiver/decoder, and that the distribution of the random numbers is the same for all samples in P .

The reconstruction as an approximation of x can readily be calculated by

$$\hat{x} = P^\dagger \hat{u} = P^\dagger \text{sign}(Px) \quad (1)$$

5

where $P^\dagger$ is the pseudo-inverse of the random matrix P .

As long as the seed for the pseudo-random columns of P is known at the decoder, the decoder can thus decode the signal from $\hat{u}$ only. Hence, in one embodiment, the seed of the pseudo-random columns are given as side-information of the encoded signal. Pseudo-random generators usually generate sequences of random values such that given the previous value $x(k)$ in the sequence, it generates the next random sample, $x(k+1) = f(x(k))$. That is, if the starting point $x(1)$ – called the "seed" value of the pseudo-random sequence – is known, then it is possible to generate the whole sequence. Hence, at the encoding and decoding side the same function for generating the random samples is used.

In the case of multiple sensor-nodes, the input signal x is assumed to be the same or noisy versions of the same signal, but each sensor has its own random matrix P_k . At the decoder the random matrices can be collated to a single large matrix $P = [P_1, P_2, \dots]$ whereby Eq. 1 remains unchanged.

It is well-known that if $K \ll N$, then P is approximately orthonormal, $P^T P \approx I$ and the quantization is near-optimal.

Here, K is not necessarily smaller than N , whereby the orthonormality becomes less accurate. Using the transpose instead of the pseudo-inverse decreases algorithmic complexity and coding efficiency, but does not impose a limitation to our experiments on perceptual modelling, since every transmitted bit still improves the accuracy of the output signal.

30

It can be expected that a source model would then be applied on the decoder side and that such a model would increase the accuracy of the reconstruction. It is however not necessary to implement a source model, since its effect can be simulated by increasing the accuracy by sending more bits.

35

The flow-diagram of an embodiment of the system (excluding the perceptual model) is illustrated in Fig. 5. Shown are n microphones capturing the audio signals from one audio source. The following encoder apparatuses 1 are in the shown embodiment part of the respective microphone- or sensor-node.

5

The n input signals 3 are transformed by the transformers 15 of the n encoding apparatuses 1 into n spectra s which are converted into n perceptually weighted spectra $x_k = W_k s_k$ which is done here by the not shown perceptual weighters.

10 The n perceptually weighted spectra x_k are submitted to the n quantizer 14.

Each quantizer 14 comprises a random matrix applicator 16 and a sign function calculator 17.

15 The random matrix applicators 16 apply a random matrix P_i being a matrix with dimensions K times N to a N times 1 vector based on the respective perceptually weighted spectrum x_i with $i = 1, 2, \dots, n$. N is an integer scalar corresponding to the number of samples in the spectrum x . K is an integer scalar corresponding to the number of rows in the randomized spectrum. By a quantization with the sign-operator, each row is

20 quantized with one bit, whereby the number of bits transmitted is K . When multiple encoders – e.g. microphones – are used, then each encoder has its own matrix P_k , which are of size $K_k \times N$. That is, each encoder sends K_k bits to the decoder, where the number of bits can vary from encoder to encoder.

25 The transformed spectra are given by: $u_i = P_i x_i$.

The sign function calculator 17 calculates the sign or sign function of the respective transformed spectrum: $\hat{u}_i = \text{sign}(u_i)$. This is done to the K components of the transformed spectra leading to K bits being transmitted via a transmission channel.

30

Fig. 6 shows one encoding apparatus 1 comprising the perceptual weighter 10 and the quantizer 14.

The audio signal 3 is transformed by a transformer 15 into a spectrum s_k . The perceptual

35 weighter 10 applies the perceptual weighting matrix W to the spectrum s_k in order to provide the perceptually weighted spectrum x_k which is quantized by the quantizer 14.

The quantizer 14 comprises the random matrix applicators 16 which receives the perceptually weighted spectrum x_k and applies the random matrix P_k to it via $P_k x_k$. The components of the resulting transformed spectrum u_k are quantized by the sign function calculator 17 via calculating the sign function of each component. This leads to a bitstreams with number of bits equaling the number of components of the transformed spectrum. The encoded signal 4 is, thus, given by the bitstream $\hat{u}_k$.

It follows a discussion of the way to decode the encoded signals.

Fig. 7 shows an embodiment of a decoding apparatus 2 for processing an encoded signal 4.

The encoded signal 4 is in form of the bitstream $\hat{u}$ de-quantized by a de-quantizer 20.

The encoded signal 4 is de-quantized by using the pseudo-inverse $P^\dagger$ of the random matrix P . The information about the random matrix is taken from the side-information of the encoded signal 4. The calculated perceptually weighted spectrum is given by: $\hat{x} = P^\dagger \hat{u}$.

The calculated perceptually weighted spectrum $\hat{x}$ is de-weighted by the perceptual de-weighter 21. The calculated spectrum $\hat{s}$ provided by the perceptual de-weighter 21 is transformed by a de-transformer 22 (e.g. via the inverse short-time Fourier transform, STFT⁻¹) into a time signal which is the extracted audio signal 5.

Fig. 8 shows an embodiment in which the de-quantizer 20 receives a plurality of encoded signals 4 from different sensor-nodes, i.e. from different microphones. The individual random matrices P_k are collated into a single large matrix $P = [P_1, P_2, \dots]$.

The single calculated perceptually weighted spectrum is then given by: $\hat{x} = \begin{bmatrix} P_1 \\ P_2 \\ \dots \\ P_n \end{bmatrix}^\dagger \begin{bmatrix} u_1 \\ u_2 \\ \dots \\ u_n \end{bmatrix}$.

In an alternative embodiment, the bit-streams are inverted with their respective random matrix $\hat{x}_k = P_k^\dagger u_k$ and the spectra are merged afterwards.

The obtained calculated perceptually weighted spectrum $\hat{x}$ is treated as described for the embodiment shown in Fig. 7.

- 5 In Fig. 9 an embodiment of the perceptual de-weighter 21 as part of a decoding apparatus is shown.

The perceptual de-weighter 21 receives from the de-quantizer 20 the calculated perceptually weighted spectrum $\hat{x}$ which is the de-quantized bitstream of the encoded
10 signal 4.

The perceptual de-weighter 21 recovers the perceptual weighted model W used during the encoding of the input signal 3 by using an iterative algorithm. Here, W_k is the k -th approximation or guess of the perceptual weighted model W .
15

The algorithm starts with an initial guess provider 23. The initial guess provider 23 provides a zeroth estimation W_0 of the perceptual weighted model W .

In the shown embodiment, the vector w_k is used which comprises the diagonal elements of the respective matrix W_k . Such a vector was used with the compression function of the
20 envelope of the magnitude spectrum during the perceptual weighting of the input signal.

Hence, a 0-th guess of the vector w_0 is given and the 0-th approximation W_0 of the perceptual weighted model W is appropriately set.
25

The initial guess of the approximation vector w_0 is set in one embodiment by using the equation used for the calculation of the envelope y of the magnitude spectrum.

So, the initial guess becomes: $w_0 = A\Lambda A^T \hat{x}$ with the matrix A and a diagonal matrix Λ .
30

The matrix A is a filterbank and the diagonal matrix Λ contains normalization factors for each band of the used filterband. The choice of matrix A is a design choice, which would typically be fixed in advance. In an embodiment, in which matrix A does not change over time, it is hardwired into both the encoder and decoder. This implies that both encoder
35 and decoder use the same known matrix A . In a different embodiment, the matrix A used for the encoding is transmitted as part of the encoded signal. In one embodiment, the

choice regarding the matrix A is transmitted in an initialization or configuration step, before the actual communication starts. The foregoing also refers to the diagonal matrix Λ .

5 The initial guess matrix W_0 is given to a spectrum approximator 24. This also starts the actual iteration with a running index k set to 0.

The spectrum approximator 24 calculates based on the data w_0 for the initial guess W_0 of the perceptual weighted model W an approximation of the spectrum.

10 This is done via applying the inverse of the initial guess matrix W_0 to a vector containing the calculated perceptually weighted spectrum $\hat{x}$.

The spectrum is approximated in this step with $k = 0$ by $\hat{s}_0 = W_0^{-1}\hat{x}$.

15 The approximated spectrum $\hat{s}_0$ is provided to a model approximator 25 that calculates a new approximation W_1 for the perceptual weighted model W.

This is based on the knowledge that the vector of the diagonal elements w_k is a function of the spectrum $\hat{s}_k$.

20

In one embodiment, the compression function $f()$, which is used during the encoding to generate the matrix W, is known at the decoding side. Like discussed before concerning the matrix A, the compression function $f()$ is either set at both sides, e.g. fixed in the respective software for encoding and decoding or is transmitted as part of the encoded signal.

25

Hence, the next approximation of the vector is calculated via: $w_1 = f(\hat{s}_0)$ and the next – here first – approximation W_1 is obtained.

30 This approximation W_1 is returned to the spectrum approximator 24 for calculating the next approximated spectrum $\hat{s}_k = W_k^{-1}\hat{x}$. The index k is increased accordingly.

This approximation serves afterwards for obtaining the next approximation W_{k+1} of the perceptual weighted model W based on the vector with diagonal elements given by the equation: $w_{k+1} = f(\hat{s}_k)$.

35

This is repeated until converged. Convergence is usually checked by comparing the current output vector $w(k)$ with the previous vector $w(k-1)$. If the difference $||w(k) - w(k-1)||$ is below some preset threshold, then the iteration has converged and iterations can be stopped.

5

The finally obtained approximated spectrum $\hat{s}$ is the output of the perceptual de-weighter 21 and is the approximation of the spectrum of the original input signal.

The reconstruction of the Perceptual Model is explained once more:

10

At the decoder side, an estimate $\hat{x}$ of the perceptual signal x (see Eq. 1) can be recovered by $\hat{x} = P^\dagger \text{sign}(Px)$ where $\hat{u} = \text{sign}(Px)$ is the quantized perceptually weighted spectrum x . This is done by the de-quantizer 20.

15

The main task is thus to recover an estimate $\hat{s}$ of the original signal s from the quantized perceptual signal $\hat{x}$.

The perceptually weighted spectrum x is based on the spectrum of the input signal via the perceptual weighted model W by the equation: $x = Ws$.

20

The aim is that the estimate equals the spectrum, i.e. $\hat{x} \approx x$.

Hence, $Ws \approx W\hat{s}$.

Furthermore, the vector w giving the diagonal elements of the matrix W are a function of the spectrum s of the input signal: $w = f(s)$.

Therefore, with an estimate of w , $\hat{s}$ can be estimated, whereby w can be estimated. This can be iterated until convergence.

25

This is thus an Expectation Maximization-type algorithm that can be described like this:

1. Get an initial guess of w_0 by, for example, $w_0 = A\Lambda A^T \hat{x}$ and set W_0 appropriately.

30

2. Repeat from $k = 0$ until converged:

(a) Calculate $\hat{s}_k = W_k^{-1} \hat{x}$.

(b) Calculate $w_{k+1} = f(\hat{s}_k)$ and set W_{k+1} appropriately.

(c) Increase k .

35

The last values $\hat{s}_k$ and W_k are the final estimates of $\hat{s}$ and $\hat{W}$.

Typically, less than 20 iterations are required for convergence

Different embodiments of the decoding apparatus 2 are also shown by Fig. 5 and Fig. 6. The decoding apparatus 2 of Fig. 5 is shown without the de-weighting. In Fig. 6 the
 5 approximation of the perceptual weighted model W is given and used for the de-weighting of the de-quantized spectrum $\hat{x}$. This highlights that the decoding includes the reconstruction of the model W based on the encoded signal.

To evaluate the performance of each part of the proposed system, the following
 10 experiments were performed.

Three versions of the input audio were compared:

the quantized and reconstructed signal, 1) without and 2) with perceptual modelling such
 15 that the perceptual is known at the decoder,
 as well as 3) the perceptually quantized signal where the reconstruction is performed with the blindly estimated perceptual model according to the invention.

As test material random speech samples from the NTT-AT dataset [21] were used (see
 20 Fig. 10 upper row). The input signals were resampled to 12.8 kHz, the STFT was implemented with discrete cosine transform to obtain a real-valued spectrum and an envelope model was used with 20 bands, distributed according to the Mel-scale [20, 5].

As a first approximation of the perceptual model, the range-reduction function of $f(y) = y^p$ with $p = 0.5$ was used. This perceptual model was chosen merely as a way to
 25 demonstrate the performance of blind reconstruction, and should not be considered as a tuned end-product. The performance of the envelope model as well as the perceptual model were already illustrated in Fig. 4.

30 First, the perceptual SNR for the quantization proposed without (SNR_O) and with blind reconstruction (SNR_B) of the perceptual model, respectively, will be estimated by:

$$SNR_O = \frac{\|x\|}{\|x-\hat{x}\|}, \text{ and } SNR_B = \frac{\|x\|}{\|x-W\hat{W}^{-1}\hat{x}\|} \quad (2)$$

Fig. 10 illustrates the perceptual SNR for a speech file quantized with the different methods ($K = 3000$).

It is clear that when the perceptual model is known (oracle approach), the SNR is close to 8.4 dB. Blind reconstruction of the perceptual model clearly decreases quality (Blind) especially for voiced phonemes. However, the SNR of the system without a perceptual model (No perc) is more than twice worse than with blind recovery.

To further quantify the advantage of blind reconstruction instead of no perceptual modelling, the mean SNR with different bit-rates K were measured (see Fig. 11).

The blind recovery and no-perceptual-model approaches are on average 1.1 dB and 5.8 dB worse than the oracle approach. Clearly SNR improves with bit-rate, though the no-perceptual-model case improves slower than with a perceptual model. Moreover, with increasing SNR, the blind recovery approaches the quality of the oracle approach asymptotically.

Finally, to evaluate subjective quality, a MUSHRA listening test was performed with eight listeners and six randomly chosen items from the NTT-AT dataset. The signal was quantized with 3 kbits/frame. That is a relatively low number given that no source modelling was used for the experiments, whereby output SNR is also relatively low. This scenario was chosen to demonstrate a problematic condition and performance is expected to improve significantly at higher bit-rates as well as when applying a source model.

From the differential MUSHRA scores in Fig. 12, it can be seen that for all items, perceptual modelling improves quality with both the oracle and blind estimation, by 29.9 and 22.3 points on average, respectively. The statistical significance of the differences were confirmed with Student's t-test at $p > 99\%$.

The proposed 1 bit quantization and coding scheme – either combined in one embodiment or in separate embodiments – have several interesting consequences and properties.

Firstly, for analyzing quantization properties, note that each column of P is a projection to a 1-dimensional sub-space of the N -dimensional space of vector x .

By encoding the sign of one projection, the N-dimensional space is split into two parts. By repeatedly encoding signs of Px , the N-dimensional space is split into ever smaller quantization cells. Since P is a random matrix, its columns are approximately orthogonal to each other, whereby the quantization cells remain near-optimal.

In a single node-system, a quantization approach may be designed which is more efficient. However, in a distributed system it gets more complicated – a simple method is needed to prevent nodes from encoding the same information, that is, over-coding has to be avoided while retaining a low algorithmic complexity. The inventive quantization is very simple and provides near-optimal performance.

Secondly, source coding methods were not employed.

It is however well-known that such modelling can be used to improve coding efficiency significantly. Source modelling can be applied at the decoder side by modelling the probability distribution of speech and audio signals (e.g. [22]). Source modelling is possible, since the quantized signal can be treated as a noisy observation of the “true” signal, whereby, by applying a prior distribution of the source, maximum likelihood optimization (or similar) can be applied to approximate the “true” signal. Since this optimization is applied in the network or at the decoder, the computational load is kept away from the sensor-nodes and the sensor-nodes can remain low-power.

Thirdly, from a privacy perspective, the random projection method can be designed to be a highly efficient encryption.

If an eavesdropper does not know the seed for the random matrix, then the data will seem entirely random and meaningless. Assuming that the random seed is communicated in a secure manner, then only the encoder and the intended receiver can decrypt a message. This approach is in contrast to approaches such as given by [12, 13], where communication between nodes is intentionally employed. While such collaboration between nodes can be used to improve perceptual SNR, privacy is more difficult to guarantee. Even when assuming that sensor-nodes are operating over a secure network, it can take only one compromised node to gain access to all communications. In the proposed approach, in contrast, if an eavesdropper gains access to one sensor node, it compromises the data of that node only, since nodes can and should use different seeds.

To limit the transmission power of sensor nodes, it can however be allowed that nodes relay packets, since packets remain readable by the intended recipient only and privacy is thus not compromised.

- 5 The inventive method is based in one embodiment on a 1 bit quantization idea, where on the encoder side, the perceptually weighted input signal is projected to random subspaces, and where the sign of each dimension is then transmitted. The decoder can invert the quantization with a pseudo-inverse, or similar, to obtain the quantized perceptually weighted signal.

10

The main part of the proposed method is then reconstruction of an estimate of the original signal, when we have access only to the perceptually weighted signal. The approach is based on an estimation-maximization (EM) algorithm, where it is iteratively alternated between estimating the perceptual model and the original signal.

15

The proposed distributed speech and audio coding algorithm is thus a viable approach for applications for the internet of things. It offers scalable performance for any number of sensor nodes and level of power consumption. Moreover, the algorithm is secure by design, since privacy of the communication channel can be guaranteed by encrypted communication of the random seed.

20

The presented invention includes at least the following features, aspects, as well as obvious applications and extensions. The listing, thus, refers to different embodiment and allows different combinations of the mentioned features.

25

1. Distributed speech and audio coding system with one or several encoders, which can be implemented on, scalable, flexible, low-power and low-cost processors with limited transmission and computational capacity and other resources.

30

- 1.1. Distributed encoders can be configured to encode information such that the information from each encoder node can independently quantized such that amount of information from a collection of nodes can be maximized, for example,

- 1.1.1. by making each encoder node encode information which is orthogonal or approximately orthogonal to the other nodes,

35

- 1.1.2. by using randomized quantization algorithms, which may or may not be orthogonal or approximately orthogonal to the other nodes,

1.1.3. by using entropy coding methods to compress quantized information.

1.2. Distributed encoders can be configured to encode information in a perceptual domain, which approximates the perceptual accuracy of the human hearing system

1.2.1. where the transformation to the perceptual domain can be designed to be reversible, such that encoder nodes can (blindly) recover the perceptual model without explicit transmission of the perceptual model,

1.2.2. where some or all parameters of the perceptual models can be explicitly transmitted from some or all encoder nodes as side information.

1.2.3. where the perceptual models can be described using envelope models, such as linear predictive models, filter banks, matrix transforms, or scale factor bands (piece-wise constant or interpolated), and they can be implemented as matrix operations or filtering operations.

1.2.4. where the perceptual models can have constant and adaptive parts, where constant parts are always the same, and adaptive parts are dependent on the input signal and/or the configuration of the system and/or the amount of resources (hardware, computation, transmission, battery capacity etc.) available.

1.3. Distributed encoders can be configured to encode information such that the quantization error is minimized in a perceptual domain, for example, like the analysis-by-synthesis loop in CELP-type codecs.

1.4. Distributed encoders can be configured to encode information using encryption to prevent eavesdropping by for example

1.4.1. encrypting the final or intermediate signal description before transmission using methods which are algorithmic or based on table-look-ups,

1.4.2. by using quantization with randomization, such as random transforms or projections, where the randomization is achieved by an algorithm or table-look-up with a known seed value (starting point).

1.5. Distributed encoders whose configuration can be flexible and modified on-line, for example,

- 1.5.1. based on location and movement of audio sources (such as human speakers), sensor nodes, as well as availability of resources, sensor nodes can decide independently or jointly which nodes are active or inactive,
- 1.5.2. based on location and movement of audio sources (such as human speakers), sensor nodes, as well as availability of resources, sensor nodes can decide independently or jointly adjust resource allocation, for example, such that a sensor node with a microphone near a speaker uses more resources than those further away.
- 1.6. A special case of the proposed system is when there is only a single encoder, whereby the system reduces to follow the classical architecture of speech and audio codecs. Importantly, however, the random quantization and blind recovery of the perceptual model -aspects of the current invention can be used also in the classical encoder&decoder design, for example, for the benefit of coding efficiency, to obtain a low-complexity encoder, and for encryption of the communication.
2. Distributed speech and audio coding system where the input from the encoders is merged at one or several network nodes (using in-network processing methods) or at one or several decoder/receiver nodes.
- 2.1. Decoder or processing units can be configured to invert quantization by for example
- 2.1.1. exact inverse transform, pseudo-inverse, or an approximated inverse such as the transpose of a random matrix,
- 2.1.2. estimating the input signal by optimization methods, such as compressive sensing,
- 2.1.3. bit-streams u_k from nodes $k \in [1, 2, \dots, n]$ can be merged by a joint-inversion such as $\hat{x} = \begin{bmatrix} P_1 \\ P_2 \\ \dots \\ P_n \end{bmatrix}^\dagger \begin{bmatrix} u_1 \\ u_2 \\ \dots \\ u_n \end{bmatrix}$ or the bit-streams can be first inverted $\hat{x}_k = P_k^\dagger u_k$ and only afterwards merged. The benefit of the latter is that we can then apply modelling, such as perceptual, source, spatial and sensor -models separately or jointly on each node.

2.2. Decoder or processing units can be configured to invert the perceptual model by using transmitted side-information and/or by blindly estimating it from the reconstructed (perceptual) signal by for example

5 2.2.1. direct methods, where the perceptual model is obtained directly from the quantized signal,

2.2.2. iterative methods, where the estimate of the perceptual model is successively improved iteration by iteration.

10 2.3. Decoder or processing units can be configured to use source and sensor-node models to improve the quality of the decoded signal.

2.3.1. By modelling statistical properties of source signals and sensor nodes, we can optimize the likelihood of observation (the quantized signal) given the prior distribution of the signal and/or source.

15 2.3.1.1. The modelled source features can include one or several spectral envelope models (linear predictive models, distribution quantization, scale factors etc.), harmonic models (comb filters, long time predictors etc.), temporal envelopes (linear predictive models, distribution quantization, scale factors etc.), spatial envelopes and spatial movement models.

20 2.3.1.2. The features can be modelled with table-lookups such as codebooks or vector quantization, or with parametric models such as physical (speech production etc.), perceptual (masking models, critical bands etc.) or statistical models (uni- or multivariate probability distributions).

25 2.3.1.3. The features can describe acoustic and spatial features of the sources, but also features of the sensor-nodes, such as signal quality (e.g. high or low SNR) and spatial features (e.g. stationary, moving, wearable, etc.).

30 2.3.2. By determining perceptually annoying distortions of the output signal, we can modify the output signal by, for example, post-filtering methods.

2.4. Decoder or processing units can be configured to use signal enhancement methods such as noise attenuation, beamforming and source separation methods to enhance signal quality.

35 2.4.1. Multimicrophone methods such as beamforming can be employed to take advantage of the spatial configuration of sensors to extract spatially

separated sound sources and other information about the room acoustics. Importantly, multimicrophone methods can typically include delay estimation and/or cross-correlation estimation.

5 2.4.2. Noise attenuation methods can be used to suppress undesirable sources such as background noise.

 2.4.3. Source separation methods can be used to distinguish between several sound sources.

10 3. Distributed speech and audio coding system where the information of the signal information can be transmitted, relayed and processed by a selection of devices/nodes in the system.

15 3.1. Sensor-nodes (devices with sensors) receive the raw signal and can be configured to either encode (quantize and code) the signal or transmit it in a raw format.

 3.2. Any node (typically the sensor-node) which has access to a raw signal can encode the signal and transmit it.

20 3.3. Any node can be configured to relay information from other nodes.

 3.4. Any node (typically a receiver node) which has access to an encoded signal (and the optional encryption-decoding information), can be configured decode the signal.

25 3.5. An intermediate node, such as server-device at the decoder side, can be configured to merge information from available streams into one or several streams. The combined stream(s) may represent for example the original acoustic representation (such as a music performance) or individual sound sources (such as individual speakers in a teleconference). The combined stream can be further reproduced by loudspeakers, stored, transmitted as such or encoded by the same or some other speech and audio coding tool.

30 3.6. The network configuration can configured to be static or dynamic, such that it optimizes for example, one or several of the following criteria; sound quality, resource allocation, security/privacy.

4. The envisioned applications include at least

5 4.1. Telephone applications, where a selection of supporting devices is used to pick up the desired speech and audio signal(s).

4.1.1. In-home and -office applications, where a selection of supporting devices are used to pick up the desired speech and audio signal(s) and transmitted to one or several remote locations.

10 4.1.2. Teleconferencing applications, where a selection of supporting devices are used to pick up the desired speech and audio signal(s) and transmitted to one or several remote locations.

4.1.3. Car-phone systems, where fixed microphones in the car and/or supporting devices within the car are used to pick up the desired speech and audio signal(s) and transmitted to one or several remote locations.

15

4.2. Gaming and virtual/augmented reality applications, where the sound scene of a player is transmitted to other players or a server.

20 4.3. Concert, performance, stage, opera, presentation, sporting, and other event production applications, where sounds of the performers, players, audience or overall sound scene is recorded or transmitted with a distributed speech and audio coding system.

4.3.1. The application can be designed to be low or ultra-low delay to enable interaction and/or simultaneous playback and amplification.

25 4.3.2. The application can be designed to allow interaction between performers, within the audience, or between all participants.

30 4.4. Safety and security applications, where a sound scene is monitored to detect dangerous events (e.g. panic at a stadium), accidents at home (e.g. an elderly person falling) etc.

4.5. Multimedia applications, where speech and audio signals are combined with video and/or other media.

35 Although some aspects have been described in the context of an apparatus, it is clear that these aspects also represent a description of the corresponding method, where a block or

device corresponds to a method step or a feature of a method step. Analogously, aspects described in the context of a method step also represent a description of a corresponding block or item or feature of a corresponding apparatus. Some or all of the method steps may be executed by (or using) a hardware apparatus, like for example, a microprocessor, a programmable computer or an electronic circuit. In some embodiments, some one or more of the most important method steps may be executed by such an apparatus.

The inventive transmitted or encoded signal can be stored on a digital storage medium or can be transmitted on a transmission medium such as a wireless transmission medium or a wired transmission medium such as the Internet.

Depending on certain implementation requirements, embodiments of the invention can be implemented in hardware or in software. The implementation can be performed using a digital storage medium, for example a floppy disc, a DVD, a Blu-Ray®, a CD, a ROM, a PROM, and EPROM, an EEPROM or a FLASH memory, having electronically readable control signals stored thereon, which cooperate (or are capable of cooperating) with a programmable computer system such that the respective method is performed. Therefore, the digital storage medium may be computer readable.

Some embodiments according to the invention comprise a data carrier having electronically readable control signals, which are capable of cooperating with a programmable computer system, such that one of the methods described herein is performed.

Generally, embodiments of the present invention can be implemented as a computer program product with a program code, the program code being operative for performing one of the methods when the computer program product runs on a computer. The program code may, for example, be stored on a machine readable carrier.

Other embodiments comprise the computer program for performing one of the methods described herein, stored on a machine readable carrier.

In other words, an embodiment of the inventive method is, therefore, a computer program having a program code for performing one of the methods described herein, when the computer program runs on a computer.

A further embodiment of the inventive method is, therefore, a data carrier (or a non-transitory storage medium such as a digital storage medium, or a computer-readable medium) comprising, recorded thereon, the computer program for performing one of the methods described herein. The data carrier, the digital storage medium or the recorded
5 medium are typically tangible and/or non-transitory.

A further embodiment of the invention method is, therefore, a data stream or a sequence of signals representing the computer program for performing one of the methods described herein. The data stream or the sequence of signals may, for example, be
10 configured to be transferred via a data communication connection, for example, via the internet.

A further embodiment comprises a processing means, for example, a computer or a programmable logic device, configured to, or adapted to, perform one of the methods
15 described herein.

A further embodiment comprises a computer having installed thereon the computer program for performing one of the methods described herein.

A further embodiment according to the invention comprises an apparatus or a system configured to transfer (for example, electronically or optically) a computer program for performing one of the methods described herein to a receiver. The receiver may, for example, be a computer, a mobile device, a memory device or the like. The apparatus or system may, for example, comprise a file server for transferring the computer program to
20 the receiver.
25

In some embodiments, a programmable logic device (for example, a field programmable gate array) may be used to perform some or all of the functionalities of the methods described herein. In some embodiments, a field programmable gate array may cooperate
30 with a microprocessor in order to perform one of the methods described herein. Generally, the methods are preferably performed by any hardware apparatus.

The above described embodiments are merely illustrative for the principles of the present invention. It is understood that modifications and variations of the arrangements and the
35 details described herein will be apparent to others skilled in the art. It is the intent, therefore, to be limited only by the scope of the impending patent claims and not by the

specific details presented by way of description and explanation of the embodiments herein.

References

5

10

15

20

25

30

35

- [1] TS 26.445, EVS Codec Detailed Algorithmic Description; 3GPP Technical Specification (Release 12), 3GPP, 2014.
- [2] TS 26.190, Adaptive Multi-Rate (AMR-WB) speech codec, 3GPP, 2007.
- [3] ISO/IEC 23003-3:2012, "MPEG-D (MPEG audio technologies), Part 3: Unified speech and audio coding," 2012.
- [4] M. Bosi, K. Brandenburg, S. Quackenbush, L. Fielder, K. Akagiri, H. Fuchs, and M. Dietz, "ISO/IEC MPEG-2 advanced audio coding," Journal of the Audio engineering society, vol. 45, no. 10, pp. 789-814, 1997.
- [5] M. Bosi and R. E. Goldberg, Introduction to Digital Audio Coding and Standards, Dordrecht, The Netherlands: Kluwer Academic Publishers, 2003.
- [6] P. T. Boufounos and R. G. Baraniuk, "1-bit compressive sensing," in Information Sciences and Systems, 2008. CISS 2008. 42nd Annual Conference on. IEEE, 2008, pp. 16-21.
- [7] Z. Xiong, A. D. Liveris, and S. Cheng, "Distributed source coding for sensor networks," IEEE Signal Process. Mag., vol. 21, no. 5, pp. 80-94, 2004.
- [8] Z. Xiong, A. D. Liveris, and Y. Yang, "Distributed source coding," Handbook on Array Processing and Sensor Networks, pp. 609-643, 2009.
- [9] B. Girod, A. M. Aaron, S. Rane, and D. Rebollo-Monedero, "Distributed video coding," Proc. IEEE, vol. 93, no. 1, pp. 71-83, 2005.
- [10] A. Majumdar, K. Ramchandran, and L. Kozintsev, "Distributed coding for wireless audio sensors," in Applications of Signal Processing to Audio and Acoustics, 2003 IEEE Workshop on. IEEE, 2003, pp. 209-212.

- 5
- [11] H. Dong, J. Lu, and Y. Sun, "Distributed audio coding in wireless sensor networks," in Computational Intelligence and Security, 2006 International Conference on, vol. 2. IEEE, 2006, pp. 1695–1699.
- [12] A. Zahedi, J. Østergaard, S. H. Jensen, P. Naylor, and S. Bech, "Coding and enhancement in wireless acoustic sensor networks," in Data Compression Conference (DCC), 2015. IEEE, 2015, pp. 293–302.
- 10 [13] A. Zahedi, J. Østergaard, S. H. Jensen, S. Bech, and P. Naylor, "Audio coding in wireless acoustic sensor networks," Signal Processing, vol. 107, pp. 141–152, 2015.
- [14] US 7,835,904.
- 15
- [15] G. Kubin and W. B. Kleijn, "Multiple-description coding (MDC) of speech with an invertible auditory model," in Speech Coding, IEEE Workshop on, 1999, pp. 81–83.
- [16] V. K. Goyal, "Multiple description coding: Compression meets the network," IEEE
- 20 Signal Process. Mag., vol. 18, no. 5, pp. 74–93, 2001.
- [17] J. O. Smith III and J. S. Abel, "Bark and ERB bilinear transforms," IEEE Trans. Speech Audio Process., vol. 7, no. 6, pp. 697–708, 1999.
- 25 [18] T. Bäckström, "Vandermonde factorization of Toeplitz matrices and applications in filtering and warping," IEEE Trans. Signal Process., vol. 61, no. 24, pp. 6257–6263, Dec. 2013.
- [19] F. Zheng, G. Zhang, and Z. Song, "Comparison of different implementations of MFCC," Journal of Computer Science and Technology, vol. 16, no. 6, pp. 582–
- 30 589, 2001.
- [20] H. Fastl and E. Zwicker, Psychoacoustics: Facts and models. Springer, 2006, vol. 22.

- [21] NTT-AT, "Super wideband stereo speech database," <http://www.ntt-at.com/product/widebandspeech>, accessed: 09.09.2014. [Online]. Available: <http://www.ntt-at.com/product/widebandspeech>
- 5 [22] S. Korse, T. Jähnel, and T. Bäckström, "Entropy coding of spectral envelopes for speech and audio coding using distribution quantization," in Proc. Interspeech, 2016.

Claims

1. Encoding apparatus for processing an input audio signal, comprising:
 - 5 a perceptual weighter; and
 - a quantizer,wherein the perceptual weighter comprises a model provider and a model applicator, wherein the model provider is configured for providing a perceptual weighting model based on the input audio signal, and wherein the model applicator is configured for providing a perceptually weighted spectrum by applying the perceptual weighting model to a spectrum based on the input audio signal,
 - 10
 - 15 wherein the quantizer is configured to quantize the perceptually weighted spectrum and for providing a bitstream, wherein the quantizer comprises a random matrix applicator and a sign function calculator, wherein the random matrix applicator is configured for applying a random matrix to the perceptually weighted spectrum in order to provide a transformed spectrum, and wherein the sign function calculator
 - 20 is configured for calculating a sign function of components of the transformed spectrum in order to provide the bitstream, andwherein the model provider is configured for calculating an envelope of a magnitude spectrum of the input audio signal, applying a compression function to the envelope of the magnitude spectrum, and calculating the perceptual weighting model based on a result of the applying the compression function.
 - 25
2. Encoding apparatus according to claim 1,
 - 30 wherein the compression function is strictly increasing,where for any positive scalar value and an arbitrarily small value a function value for the positive scalar value is smaller than the function value for the sum of the positive scalar value and of the arbitrarily small value, and
 - 35

wherein for a first positive scalar value and a second positive scalar value greater than the first positive scalar value, a difference between the function value for the second positive scalar value and the function value for the first positive scalar value is smaller than a difference between the second positive scalar value and the first positive scalar value.

3. Decoding apparatus for processing an encoded audio signal, comprising:

a de-quantizer; and

a perceptual de-weighter,

wherein the de-quantizer is configured to de-quantize a bitstream comprised by the encoded audio signal and for providing a calculated perceptually weighted spectrum, wherein the de-quantizer is configured to de-quantize the bitstream by applying a pseudo-inverse of a random matrix to the bitstream,

wherein the perceptual de-weighter comprises a spectrum approximator and a model approximator, wherein the spectrum approximator is configured for calculating an approximation of a spectrum based on the calculated perceptually weighted spectrum, and wherein the model approximator is configured for calculating an approximation of a perceptual weighting model with which the encoded audio signal is associated based on the approximation of the spectrum,

wherein the perceptual de-weighter comprises an initial guess provider,

wherein the initial guess provider is configured for providing data for an initial guess of the perceptual weighting model, and

wherein the spectrum approximator is configured for iteratively calculating the approximation of the spectrum based on the calculated perceptually weighted spectrum and the initial guess of the perceptual weighting model or the approximation of the perceptual weighting model.

4. Decoding apparatus according to claim 3, wherein the spectrum approximator is configured for

calculating an initial approximation of the spectrum based on the initial guess of the perceptual weighting model and the calculated perceptually weighted spectrum and calculating a first approximation of the perceptual weighting model using the initial approximation of the spectrum, and

calculating a first approximation of the spectrum based on the first approximation of the perceptual weighting model and the calculated perceptually weighted spectrum and calculating a second approximation of the perceptual weighting model using the first approximation of the spectrum.

5. Decoding apparatus according to claim 4, wherein the spectrum approximator is configured for

calculating the first approximation of the perceptual weighting model using a calculation of an envelope estimate of the initial approximation of the spectrum, or calculating the second approximation of the perceptual weighting model using a calculation of an envelope estimate of the first approximation of the spectrum.

6. Decoding apparatus according to claim 5, wherein the spectrum approximator is configured for

calculating the envelope estimate of the initial approximation of the spectrum using calculating a spectral envelope of the initial approximation of the spectrum and applying a compression function to the spectral envelope of the initial approximation of the spectrum, or calculating the envelope estimate of the first approximation of the spectrum using calculating a spectral envelope of the first approximation of the spectrum and applying a compression function to the spectral envelope of the first approximation of the spectrum.

7. Decoding apparatus according to claim 3, wherein the spectrum approximator is configured for performing the iterative calculation of the approximation of the spectrum until a convergence criterion is fulfilled, wherein the approximation of the spectrum obtained in a last iteration step k is a decoded audio signal.

8. Decoding apparatus according to claim 7, wherein the spectrum approximator is configured for checking, as the convergence criterion, a result of a comparison of an output vector $w(k)$ of a current iteration step k and an output vector $w(k-1)$ of a preceding iteration step $k-1$.
- 5 9. Decoding apparatus according to claim 8, wherein the spectrum approximator is configured for determining a convergence of the iterative calculation, when a difference between the output vector $w(k)$ of the current iteration step k and the output vector $w(k-1)$ of the preceding iteration step $k-1$ is below a preset threshold.
- 10 10. Decoding apparatus according to any one of claims 8 or 9, wherein the spectrum approximator is configured for calculating, as the output vector, an envelope estimate of an approximation of the spectrum in the current iteration step k and an envelope estimate of an approximation of the spectrum in the preceding iteration step $k-1$.
- 15 11. Decoding apparatus according to any one of claims 3 to 10, wherein the de-quantizer is configured to de-quantize the bitstream based on side-information about the random matrix comprised by the encoded audio signal.
- 20 12. Decoding apparatus according to any one of claims 3 to 11, wherein the de-quantizer is configured to receive a plurality of bitstreams and to provide the calculated perceptually weighted spectrum based on the plurality of bitstreams.
- 25 13. Decoding apparatus according to any one of claims 5 to 12, wherein the decoding apparatus is configured to apply source modelling.
14. Method for processing an input audio signal, comprising:
 - 30 providing a perceptual weighting model based on the input audio signal,
 - weighting a spectrum of the input audio signal by applying the perceptual weighting model to the spectrum of the input audio signal, and
 - 35 quantizing the weighted spectrum by calculating a sign function of random projections of the weighted spectrum,

wherein the providing the perceptual weighting model comprises calculating an envelope of a magnitude spectrum of the input audio signal, applying a compression function to the envelope of the magnitude spectrum, and calculating the perceptual weighting model based on a result of the applying.

15. Method according to claim 14, further comprising:

obtaining the random projections of the weighted spectrum by applying a random matrix to the weighted spectrum.

16. Method according to claim 15, further comprising:

obtaining the envelope by using a filterbank comprising a plurality of frequency bands and a diagonal matrix containing normalization factors for each frequency band.

17. Method for processing an encoded audio signal, comprising:

providing a quantized perceptual signal by applying a pseudo-inverse of a random matrix to the encoded audio signal,

calculating an estimation of a spectrum based on the quantized perceptual signal, and

calculating an approximation of a perceptual weighting model used for providing the encoded audio signal based on the estimation of the spectrum,

wherein the calculating the approximation of the perceptual weighting model comprises providing data for an initial guess of the perceptual weighting model, and

wherein the calculating the estimation of the spectrum comprises iteratively calculating the estimation of the spectrum based on a perceptually weighted spectrum and the initial guess of the perceptual weighting model or the approximation of the perceptual weighting model.

18. Method according to claim 17, wherein the calculating the estimation of the spectrum comprises calculating an initial approximation of the spectrum based on the initial guess of the perceptual weighting model.

5

19. Method according to claim 17, further comprising:

obtaining the data for the initial guess of the perceptual weighting model by using a filterbank and a diagonal matrix containing normalization factors for each band.

10

20. A computer-readable medium having computer-readable code stored thereon to perform the method according to any one of claims 14 to 19 when the computer-readable code is run by a computer.

1/10

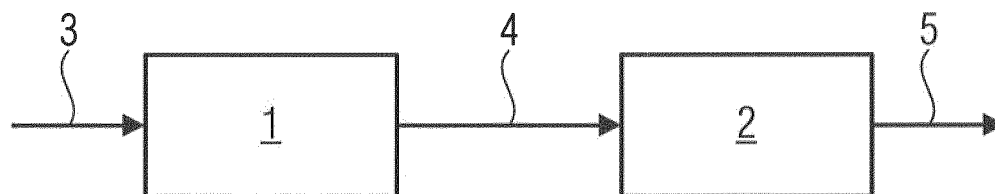


Fig. 1

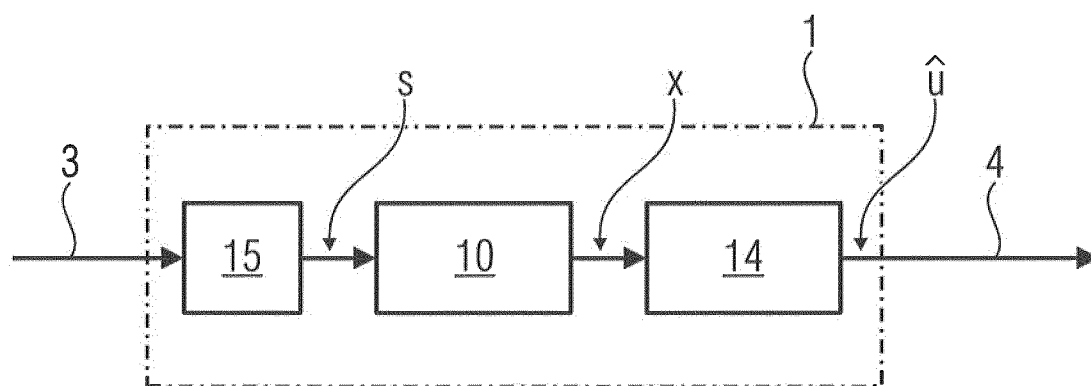


Fig. 2

2/10

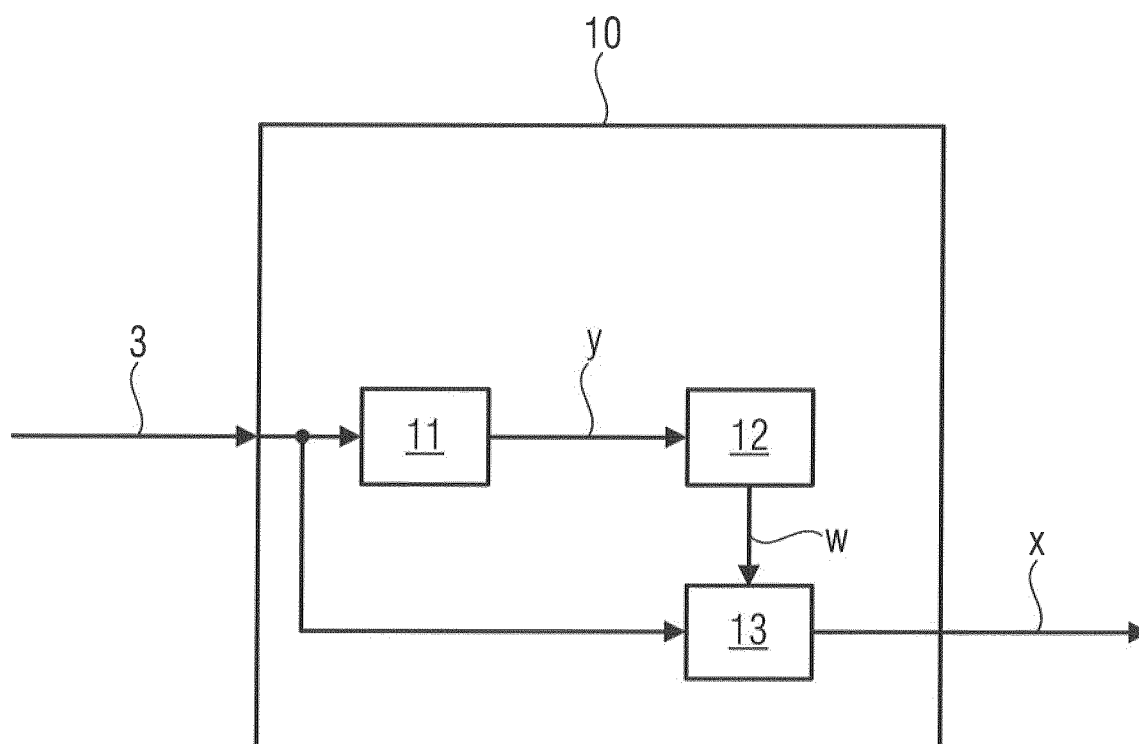


Fig. 3

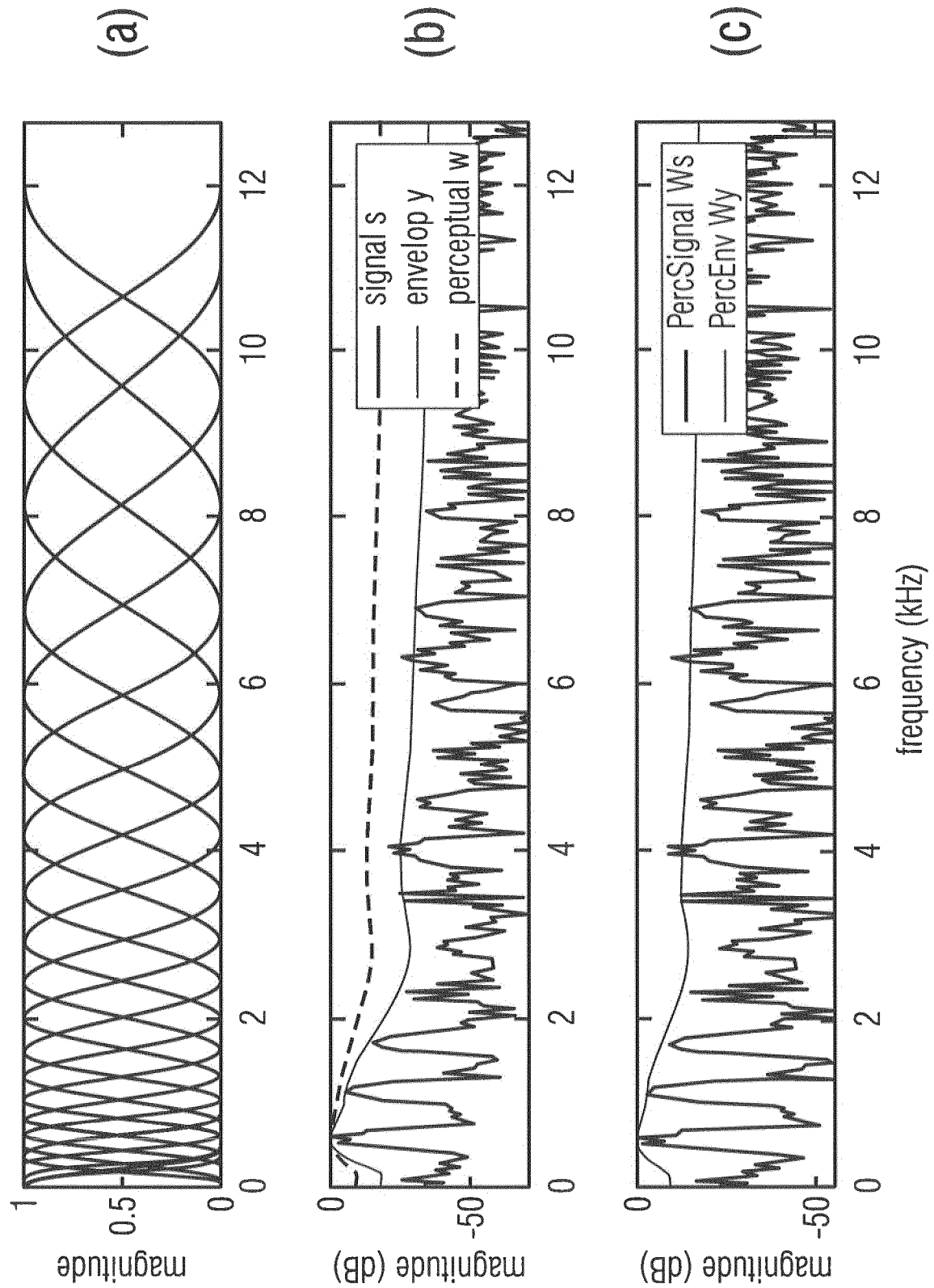


Fig. 4

4/10

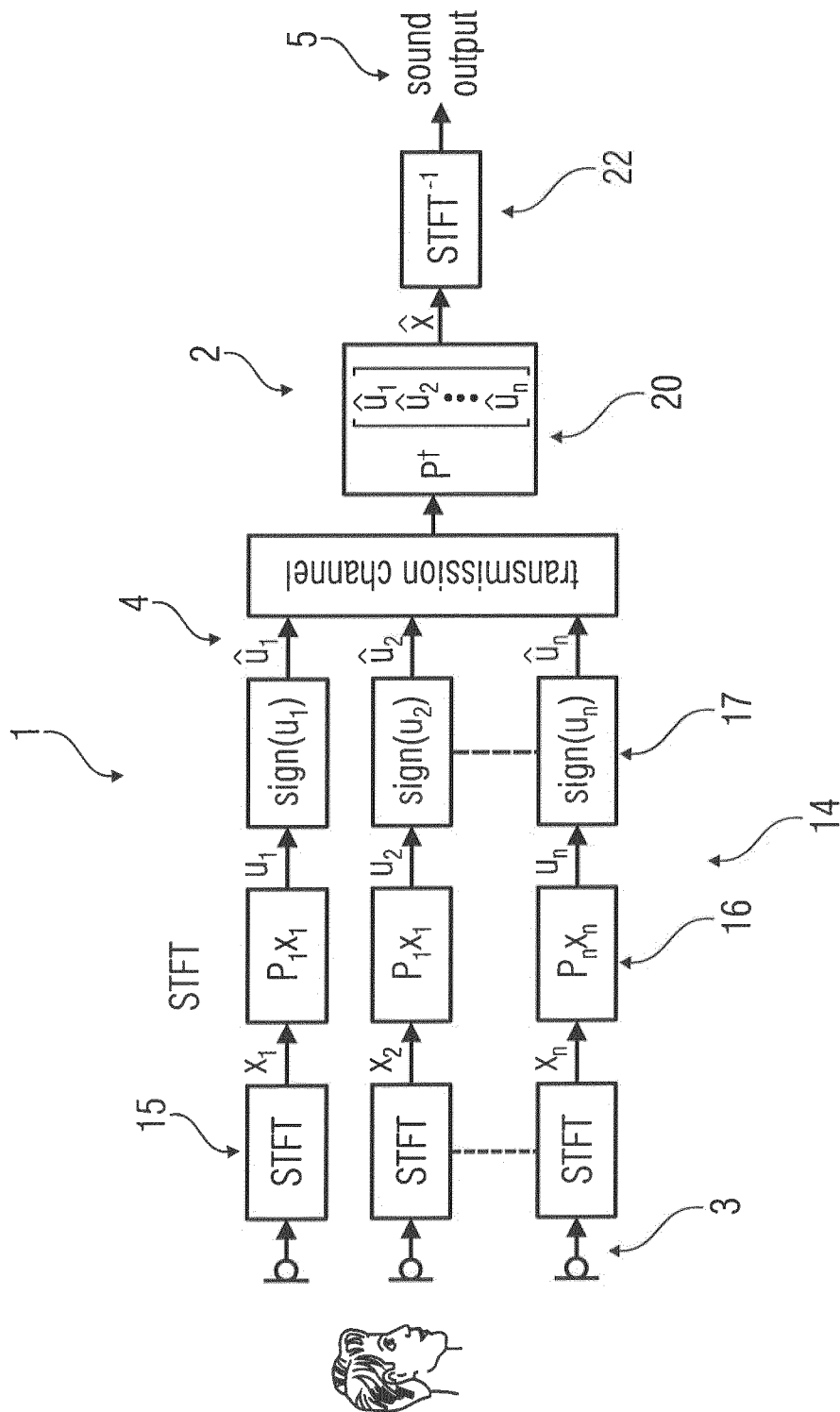


Fig. 5

5/10

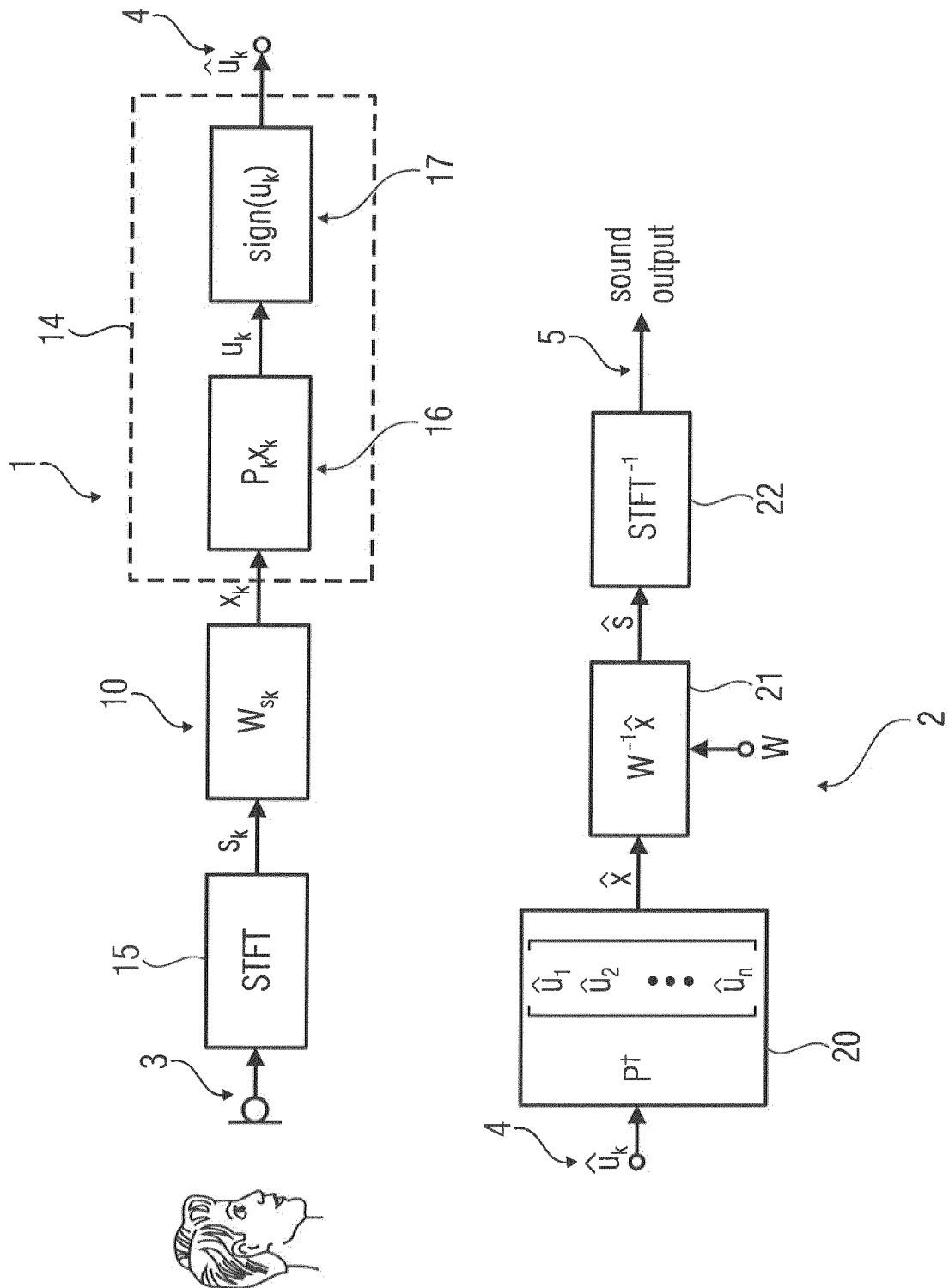


Fig. 6

6/10

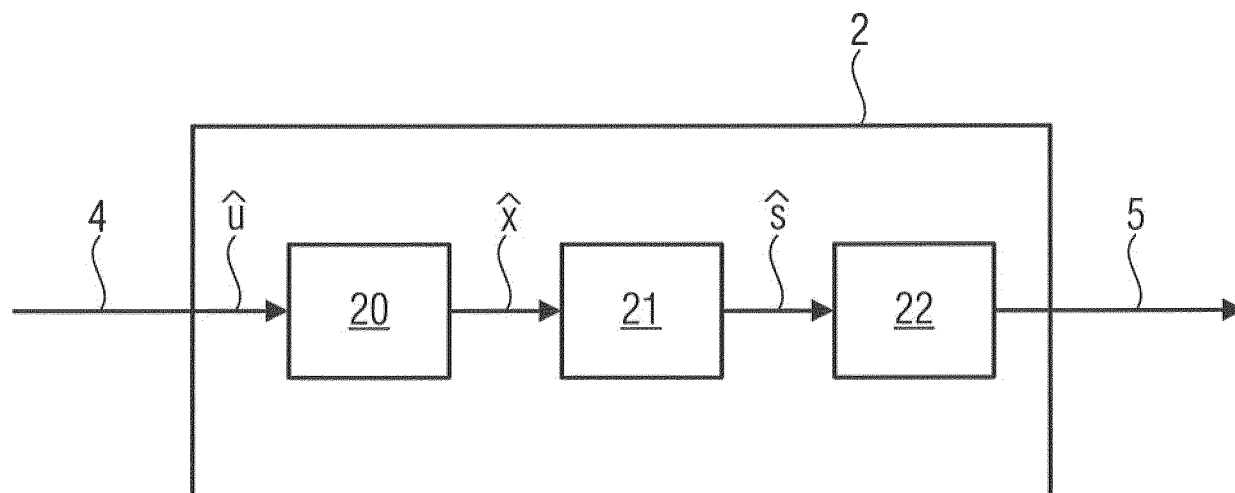


Fig. 7

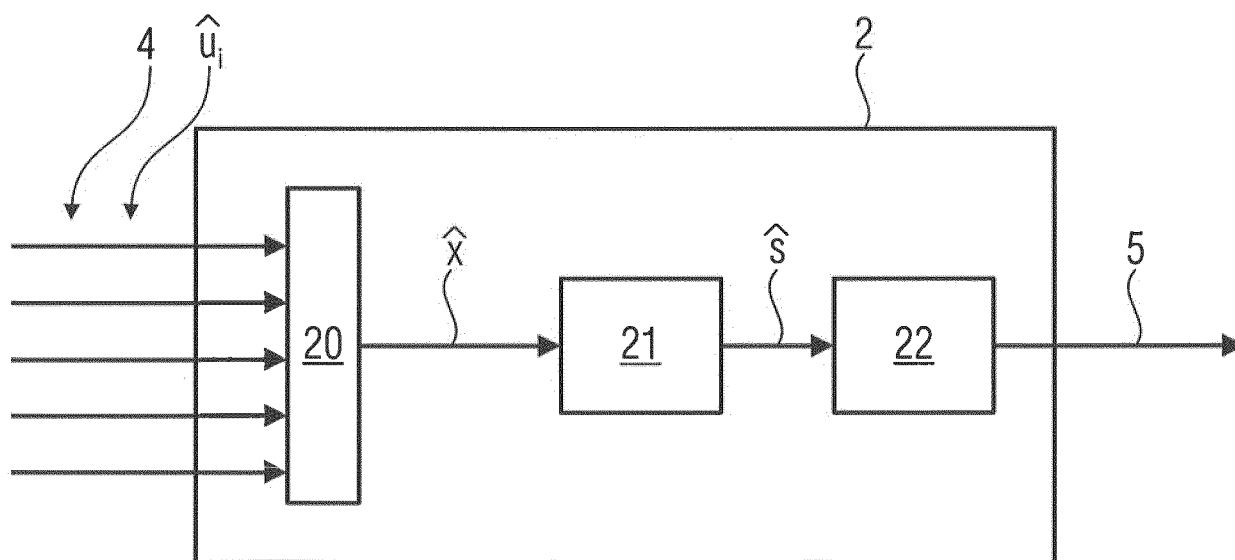


Fig. 8

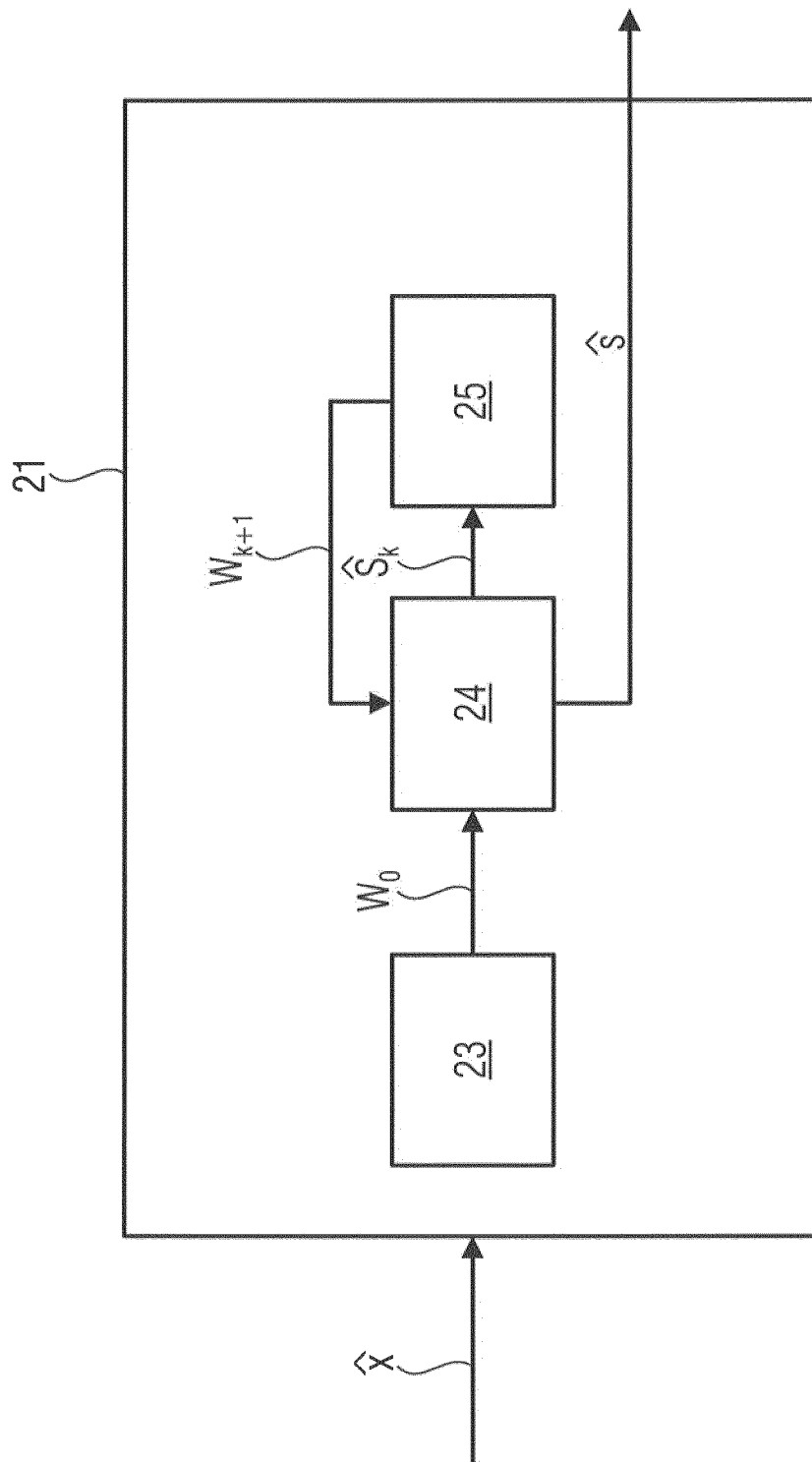


Fig. 9

8/10

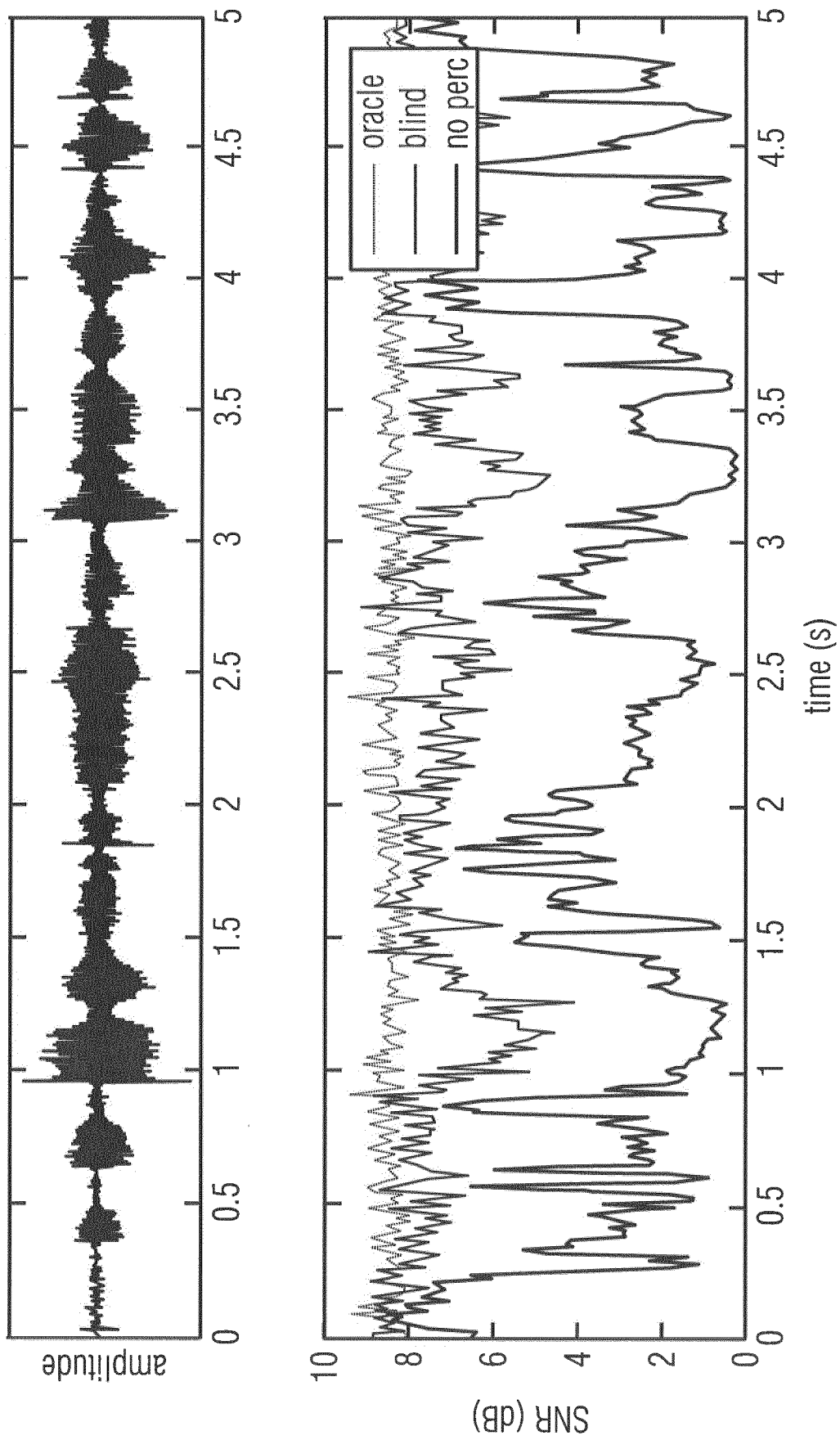


Fig. 10

9/10

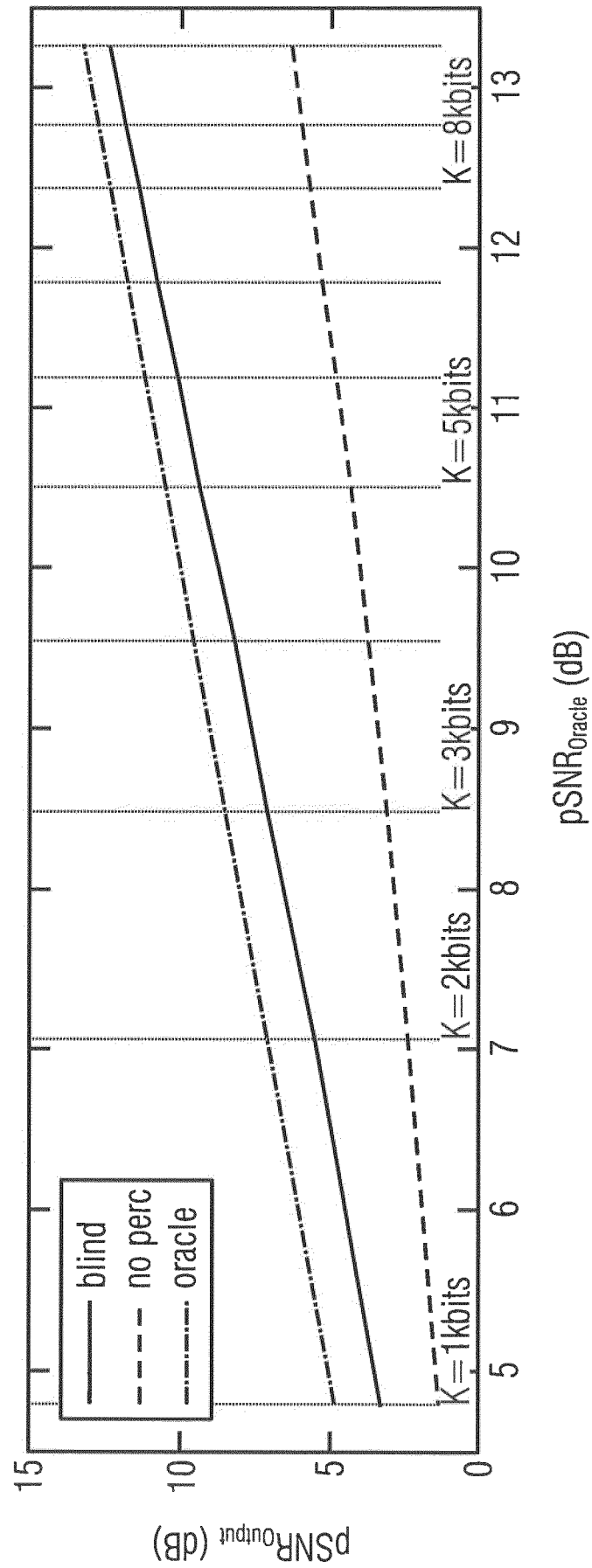


Fig. 11

10/10

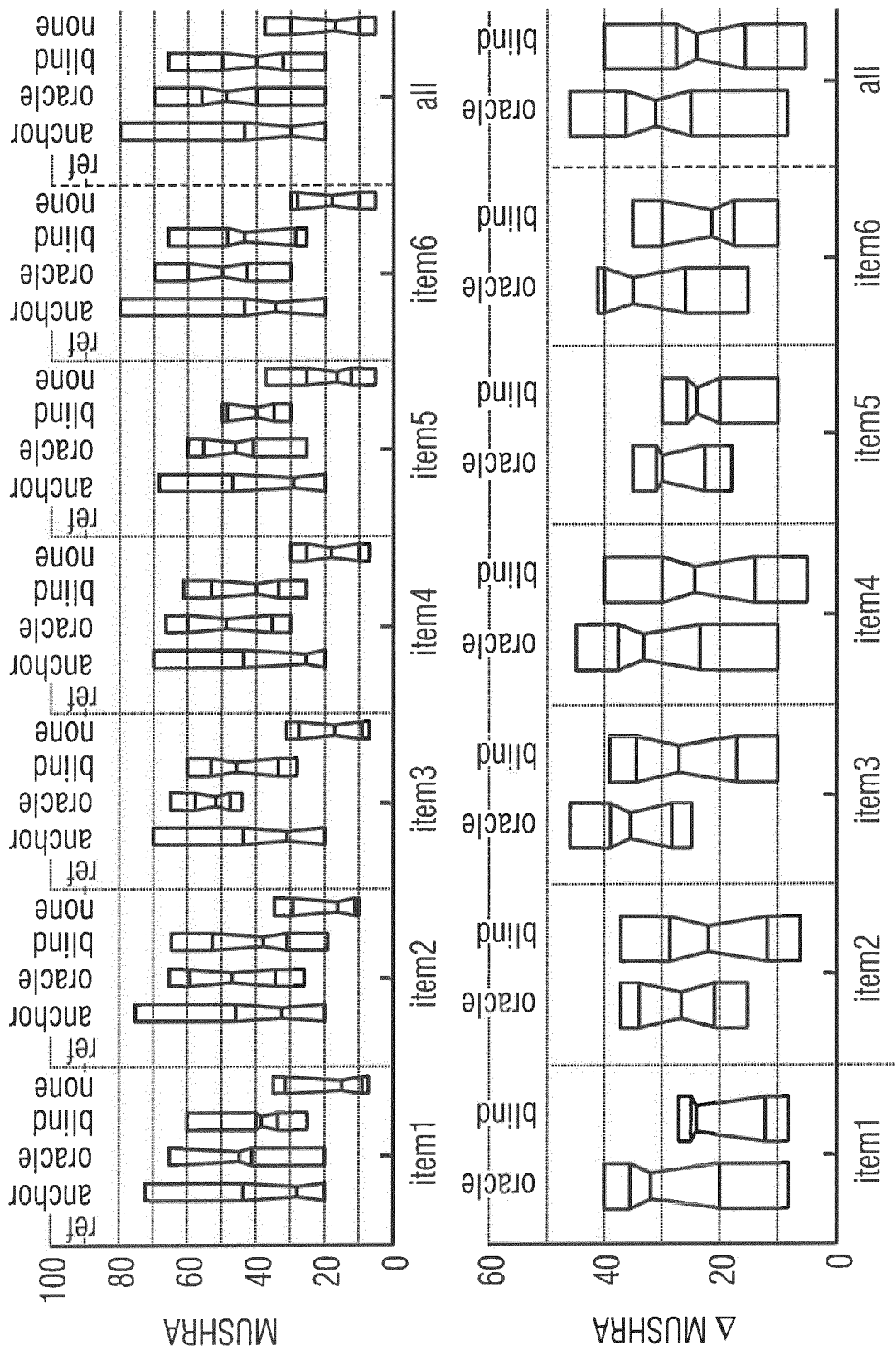


Fig. 12

